

TRABAJO FIN DE GRADO

Grado en Ingeniería Electrónica Industrial y Automática

ESTUDIO, PROGRAMACIÓN Y SIMULACIÓN 3D DE UNA CELDA ROBOTIZADA DE CLIPAJE.



Memoria

Autor: Zaharia, Vlad
Director: Ponsa Asensio, Pere
Convocatoria: junio 2019

Resumen

En el presente trabajo se describen los pasos a seguir para desarrollar una celda robotizada para una aplicación en el sector automovilístico. La función principal de la instalación es la de ensamblar clips en el interior de una pieza de plástico de mayor tamaño que pertenece al interior de una puerta de automóvil. Para llevar a cabo esta acción se utiliza un robot articulado industrial IRB 2600, que manipula la pieza de automóvil e introduce clips en ella.

Las partes del proyecto que más relevancia se les atribuye en este documento son el estudio previo de la solución, la creación de la simulación 3D mediante el programa RobotStudio, la programación del controlador del robot mediante el uso del lenguaje RAPID y la programación del sistema de visión artificial de la marca KEYENCE. También se da mucha importancia a la gestión y toma de decisiones en el proyecto, además de las modificaciones y la evolución que este ha sufrido. Asimismo, se contempla la normativa que afecta a este tipo de instalaciones, con el fin de cumplirla estrictamente.

Finalmente, la puesta en marcha de la instalación en la propia planta del cliente ha resultado satisfactoria, y esta se encuentra actualmente en producción.

El presente documento también tiene la función final de servir como guía para futuros proyectos de características similares.

Última modificación de este documento: 1 de junio de 2019

Resum

En el present treball es descriuen els passos a seguir per desenvolupar una cel·la robotitzada per a una aplicació en el sector automobilístic. La funció principal de la instal·lació és la d'acoblar clips a l'interior d'una peça de plàstic de major grandària que pertany a l'interior d'una porta d'automòbil. Per dur a terme aquesta acció s'utilitza un robot articulat industrial IRB 2600, que manipula la peça d'automòbil i introdueix clips en ella.

Les parts del projecte que més rellevància se'ls atribueix en aquest document són l'estudi previ de la solució, la creació de la simulació 3D mitjançant el programa RobotStudio, la programació del controlador del robot mitjançant l'ús del llenguatge RAPID i la programació del sistema de visió artificial de la marca KEYENCE. També es dona molta importància a la gestió i presa de decisions en el projecte, a més de les modificacions i l'evolució que aquest ha sofert. Així mateix, es contempla la normativa que afecta aquest tipus d'instal·lacions, per tal de complir-la estrictament.

Finalment, la posada en marxa de la instal·lació en la pròpia planta del client ha resultat satisfactòria, i aquesta es troba actualment en producció.

El present document també té la funció final de servir com a guia per a futurs projectes de característiques similars.

Abstract

In the present work there are described the steps to follow to develop a robotic cell for an application in the automotive sector. The main function of the installation is to assemble clips inside a larger piece of plastic that belongs to the inside of a car door. To carry out this action an industrial articulated robot IRB 2600 is used, which manipulates the car part and introduces clips into it.

The parts of the project that are most important in this document are the previous study of the solution, the creation of 3D simulation through the RobotStudio program, the programming of the robot controller using the RAPID language and the programming of the artificial vision of the KEYENCE brand. There is also a lot of importance given to the management and decision making in the project, in addition to the modifications and the evolution that this has undergone. Likewise, the regulations that affect this type of facility are contemplated, in order to strictly comply with it.

Finally, the commissioning of the installation in the customer's own plant has been satisfactory, and it is currently in production.

This document also has the final function of serving as a guide for future projects with similar characteristics.

Agradecimientos

En primer lugar, me gustaría agradecer toda la ayuda y apoyo recibido por parte del equipo del proyecto de ABB. Especialmente al Project Manager del equipo de proyectos de ingeniería, quien me facilitó todos los medios y recursos que necesité durante el desarrollo del proyecto. Así mismo, agradezco la posibilidad que me fue brindada para realizar las formaciones impartidas por ABB University Spain.

También deseo agradecer a Pere Ponsa Asensio, quien fue director de este Trabajo de Fin de Grado, su contribución al desarrollo del proyecto.

Por último, pero no menos importante, agradecerle a mi amigo y compañero Carles Treviño Mascaró la ayuda con el desarrollo del modelo 3D en realidad aumentada, además de los consejos sabiamente otorgados.

Glosario

A continuación, se definirán ciertos términos mencionados en el presente trabajo, para la mejora de la comprensión de los textos escritos.

Robot articulado: Un robot articulado es un manipulador multifuncional, reprogramable y automático con tres o más grados de libertad y que tanto puede ser fijo como móvil.^[1]

Unidad mecánica: Es el elemento del sistema, encargado de realizar las tareas de manipulación o movimiento para las que ha sido programado. En las celdas robotizadas de clipaje suele referirse a los robots o las mesas rotatorias.

Clips: Referido a las piezas pequeñas, tanto metálicas como plásticas, usadas en la industria automovilística, que se colocan en piezas plásticas de mayor tamaño (tablero de mandos, interior de las puertas, etc.) con la función de unir estas piezas de forma mecánica.

Clipaje: Termino usado en la industria automovilística para referirse a la acción de colocar clips en las piezas de plástico de mayor tamaño.

Uillaje o Útil (“Nest” en inglés): En el contexto de la industria automovilística, referido a la estructura mecánica, habitualmente hecha de polímeros, que tiene la función de sostener piezas de plástico, mientras estas reciben alguna operación.

Alimentador (“Feeder” en inglés): En el contexto de la presente memoria, se refiere la máquina que suministra clips de forma continua al acceso del robot.



Índice

| | |
|---|------------|
| RESUMEN | I |
| RESUM | II |
| ABSTRACT | III |
| AGRADECIMIENTOS | IV |
| GLOSARIO | V |
| PREFACIO | 15 |
| 0.1. Origen del trabajo | 15 |
| 0.2. Motivación | 16 |
| 0.3. Requisitos previos | 16 |
| INTRODUCCIÓN | 17 |
| 0.4. Objetivos del trabajo..... | 17 |
| 0.5. Alcance del trabajo | 18 |
| CAPÍTULO I | 20 |
| 1.1. Antecedentes | 20 |
| 1.1.1. Robótica industrial..... | 20 |
| 1.2. Conceptos básicos de los sistemas robóticos..... | 22 |
| 1.2.1. Sistema de coordenadas | 22 |
| 1.2.2. Herramienta y “TCP” (Tool Center Point) | 23 |
| 1.2.3. Objeto de trabajo | 25 |
| 1.2.4. Movimiento | 26 |
| 1.2.5. RobotStudio..... | 27 |
| 1.2.6. Visión artificial (KEYENCE) | 28 |
| CAPÍTULO II | 30 |
| 2.1. Requisitos previos y estudio de alternativas | 30 |
| 2.2. Metodología de desarrollo de proyecto..... | 33 |
| 2.2.1. Planificación..... | 36 |
| 2.3. La celda y sus componentes | 37 |
| 2.3.1. Componentes de la celda | 38 |
| 2.4. Elementos principales de la máquina..... | 39 |
| 2.4.1. Robot IRB 2600-20/1.65 | 39 |

| | | |
|---------|--|----|
| 2.4.2. | Controlador IRC 5 | 41 |
| 2.4.3. | Consola de programación (FlexPendant) | 43 |
| 2.4.4. | Armario eléctrico | 44 |
| 2.4.5. | PLC Siemens SIMATIC S7-1515F | 45 |
| 2.4.6. | Cámara de visión artificial..... | 46 |
| 2.4.7. | Utillaje (Nido) | 47 |
| 2.4.8. | Herramientas | 48 |
| 2.4.9. | Alimentadores de clips..... | 50 |
| 2.4.10. | Sistema de limpieza | 51 |
| 2.4.11. | Sistema de autoeyección | 51 |
| 2.4.12. | Pantalla HMI..... | 52 |
| 2.5. | Elementos de seguridad | 53 |
| 2.5.1. | Seguridad PLC Siemens S7-1515F..... | 53 |
| 2.5.2. | Seguridad del robot IRB 2600 y el controlador IRC 5 | 54 |
| 2.5.3. | Vallado..... | 55 |
| 2.5.4. | Puertas y cerrojos | 55 |
| 2.6. | Ciclo de funcionamiento de la máquina..... | 57 |
| 2.7. | La simulación 3D | 59 |
| 2.7.1. | Creación de la estación virtual..... | 59 |
| 2.7.2. | Preparación de la simulación | 60 |
| 2.7.3. | Creación de las trayectorias..... | 62 |
| 2.8. | Programación básica de la simulación | 63 |
| 2.8.1. | Creación de ejecutables..... | 67 |
| 2.9. | Estudio del tiempo de ciclo | 68 |
| 2.10. | Desarrollo de la programación del controlador IRC 5..... | 69 |
| 2.10.1. | Guía de programación | 70 |
| 2.10.2. | Primeros pasos..... | 71 |
| 2.10.3. | Explicación de las tareas | 73 |
| 2.10.4. | Explicación de los módulos..... | 74 |
| 2.10.5. | Explicación de las rutinas | 77 |
| 2.10.6. | Zonas mundo (World Zones) | 80 |
| 2.10.7. | Software de seguridad (SafeMove 2 Pro Virtual)..... | 81 |
| 2.10.8. | Interacción con periféricos | 83 |
| 2.10.9. | Comunicación con PLC y Máquina de soldadura | 84 |
| 2.11. | Programación del sistema de visión artificial..... | 86 |
| 2.12. | Modificaciones realizadas en el diseño..... | 93 |
| 2.13. | Comparación del tiempo de ciclo y optimización del proceso | 96 |

| | |
|---|------------|
| CAPÍTULO III | 97 |
| 3.1. Aprobación del cliente | 97 |
| 3.2. Resultado final del proyecto | 98 |
| PRESUPUESTO Y ANÁLISIS ECONÓMICO | 99 |
| ANÁLISIS DEL IMPACTO AMBIENTAL | 100 |
| CONCLUSIONES | 101 |
| REFERENCIAS BIBLIOGRÁFICAS | 103 |

Índice de figuras

| | |
|--|----|
| Figura 1.1. Imagen representativa de la aplicación de robots industriales en el sector automovilístico (Fuente: The Robot Report)..... | 21 |
| Figura 1.2. Figura que representa el sistema de coordenadas de la base del robot IRB 2600 (Fuente: Manual de instrucciones del IRB 2600)..... | 23 |
| Figura 1.3. Captura representativa del TCP de la herramienta tChange (Fuente: Simulación Vlad Zaharia) | 24 |
| Figura 1.4. Captura representativa de los 10 TCPs de la herramienta tB (Fuente: Simulación Vlad Zaharia) | 24 |
| Figura 1.5. Captura representativa de los 5 TCPs de la herramienta tA (Fuente: Simulación Vlad Zaharia) | 24 |
| Figura 1.6. Imagen del objeto de trabajo de clipaje donde se coloca la pieza a manipular (Fuente: Vlad Zaharia) | 25 |
| Figura 1.7. y Figura 1.8. Situación de los tres puntos ortogonales que definen un objeto de trabajo (izquierda) y punto metálica usada para obtener mayor precisión en la declaración de objetos de trabajo (derecha) (Fuente: Vlad Zaharia) | 25 |
| Figura 1.9. Imagen que muestra la situación de los ejes del robot, junto a su nomenclatura y el sentido de giro (Fuente: Manual del robot IRB 2600)..... | 27 |
| Figura 1.10. Ejemplo de captura de la interfaz del Terminal Software de KEYENCE (Fuente: Vlad Zaharia) | 29 |
| Figura 2.1. y Figura 2.2. Medidas de la celda final desde la vista frontal (Izquierda) y lateral (Derecha) (Fuente: Plano de “LayOut” de la celda) | 30 |
| Figura 2.3. Captura del modelo 3D de la primera solución presentada (Fuente: Vlad Zaharia) | 32 |
| Figura 2.4. Croquis de la solución inicial (Fuente: Vlad Zaharia)..... | 32 |
| Figura 2.5. Diagrama de flujo del desarrollo del proyecto. (Fuente: Guía para elaboración de proyectos) | 35 |

| | |
|--|----|
| Figura 2.6. Diagrama de Gantt del desarrollo del proyecto. (Fuente: Guía para elaboración de proyectos)..... | 36 |
| Figura 2.7. La estación 3D de la instalación (Fuente: Vlad Zaharia) | 37 |
| Figura 2.8. Imagen del robot IRB 2600-20/1.65 (Fuente: Manual de usuario IRB 2600)..... | 39 |
| Figura 2.9. Representación del área de trabajo del IRB 2600-20/1.65 (Fuente: Manual de usuario IRB 2600)..... | 40 |
| Figura 2.10. Distribución de los distintos módulos y unidades que forman el IRC 5 (Fuente: Vlad Zaharia) | 41 |
| Figura 2.11. Imagen de la FlexPendant y la situación de sus principales elementos. (Fuente: Manual de instrucciones IRC 5) | 43 |
| Figura 2.12. Fotografía del interior del armario eléctrico. (Fuente: Vlad Zaharia)..... | 44 |
| Figura 2.13. Imagen del PLC Siemens S7-1515F. (Fuente: Vlad Zaharia) | 45 |
| Figura 2.14. Imagen de la cámara CV-200M con la óptica CA-LH8. (Fuente: Manual de usuario CV-200M) | 46 |
| Figura 2.15. Fotografía del soporte de la cámara CV-200M. (Fuente: Vlad Zaharia)..... | 46 |
| Figura 2.16. y Figura 2.17. Fotografía del utillaje de la estación de clipaje vacío (Izquierda) y junto con la pinza de manipulación y la pieza (Derecha). (Fuente: Vlad Zaharia)..... | 47 |
| Figura 2.18. Fotografía del utillaje de la máquina de soldadura. (Fuente: Vlad Zaharia) | 47 |
| Figura 2.19. Fotografía del sistema de cambio de herramienta SCHUNK. (Fuente: Vlad Zaharia)..... | 48 |
| Figura 2.20. Fotografía de las pinzas de la herramienta de manipulación. (Fuente: Vlad Zaharia)..... | 48 |
| Figura 2.21. y Figura 2.22. Captura del modelo 3D (Izquierda) y fotografía (Derecha) de la herramienta de clipaje (Fuente: Vlad Zaharia)..... | 49 |
| Figura 2.23 y Figura 2.24. Captura del modelo 3D del clip (Izquierda). Fotografía de un clip (Derecha). (Fuente: Vlad Zaharia) | 49 |
| Figura 2.25. y Figura 2.26. Modelo 3D del alimentador de clips (Izquierda) y fotografía del interior del tambor de vibración. (Fuente: Vlad Zaharia)..... | 50 |

| | |
|--|----|
| Figura 2.27. Fotografía del dispensador lineal de alimentación de clips. (Fuente: Vlad Zaharia) | 50 |
| Figura 2.28. Fotografía de la pinza de limpieza de clips. (Fuente: Vlad Zaharia) | 51 |
| Figura 2.29. Fotografía del sistema de autoeyección. (Fuente: Vlad Zaharia) | 51 |
| Figura 2.30. Fotografía del sistema Up&Down y la pantalla táctil igual a la usada en esta estación. (Fuente: Vlad Zaharia) | 52 |
| Figura 2.31. Imagen del cerrojo instalado en las puertas de acceso. (Fuente: Catálogo del fabricante) | 56 |
| Figura 2.32. y Figura 2.33. Comparación entre la estación 3D inicial (Derecha) y la final (Izquierda). (Fuente: Vlad Zaharia) | 60 |
| Figura 2.34. Captura del modelo 3D del paquete energético acoplado al eje 3 del IRB 2600. (Fuente: Vlad Zaharia) | 61 |
| Figura 2.35. Captura del modelo 3D del cambio de herramienta y la herramienta de manipulación acoplados al eje 6 del IRB 2600. (Fuente: Vlad Zaharia) | 61 |
| Figura 2.36. Captura de las trayectorias (amarillo) realizadas por el robot IRB 2600 en la estación virtual. (Fuente: Vlad Zaharia) | 62 |
| Figura 2.37. Interfaz del ejecutable de la simulación. (Fuente: Vlad Zaharia) | 67 |
| Figura 2.38. Captura del estudio del tiempo de ciclo obtenido de la simulación. (Fuente: Vlad Zaharia) | 68 |
| Figura 2.39. Estructura del programa RAPID. (Fuente: Vlad Zaharia) | 74 |
| Figura 2.40. Contenido del módulo Change_Tool. (Fuente: Vlad Zaharia) | 75 |
| Figura 2.41. Contenido del módulo Drop_Pick_Part. (Fuente: Vlad Zaharia) | 75 |
| Figura 2.42. Contenido del módulo SYSCLIPPING. (Fuente: Vlad Zaharia) | 76 |
| Figura 2.43. Contenido del módulo WZ_R1. (Fuente: Vlad Zaharia) | 76 |
| Figura 2.44. Captura del modelo 3D del vallado virtual (Derecha) y las medidas de esta área (Izquierda). (Fuente: Vlad Zaharia) | 82 |

| | |
|--|----|
| Figura 2.45. Captura del modelo 3D de las herramientas virtuales (Derecha) y las medidas de esta área (Izquierda). (Fuente: Vlad Zaharia) | 83 |
| Figura 2.46. GRAFCET de los pasos más relevantes de la comunicación entre el IRC 5 y la máquina de soldadura. (Fuente: Vlad Zaharia) | 85 |
| Figura 2.47. Resumen de las características del sistema de visión artificial (Fuente: Proveedor oficial KEYENCE) | 86 |
| Figura 2.48. Menú de herramientas del controlador CV-X150RP (Fuente: Vlad Zaharia) | 88 |
| Figura 2.49. Configuración de la herramienta de detección de presencia por área blanca (Fuente: Vlad Zaharia)..... | 89 |
| Figura 2.50. Configuración de la herramienta de detección de presencia por patrón de sombras. (Fuente: Vlad Zaharia) | 90 |
| Figura 2.51. Señales de envío del controlador CV-X150RP al PLC mediante PROFINET en caso de juicio OK de todas las herramientas. (Fuente: Vlad Zaharia) | 91 |
| Figura 2.52. Captura de la pantalla de ejecución del controlador CV-X150RP donde se observan los juicios de las herramientas. (Fuente: Vlad Zaharia) | 92 |
| Figura 2.53. y Figura 2.54. Comparación entre la posición inicial (Izquierda) y final (Derecha) del utillaje. (Fuente: Vlad Zaharia) | 93 |
| Figura 2.55 y Figura 2.56. Captura del modelo 3D (Izquierda) y fotografía (Derecha) del soporte de la cámara. (Fuente: Vlad Zaharia)..... | 94 |
| Figura 2.57. Fotografía del soporte (raqueta) del paquete energético situado en el eje 6 del robot. (Fuente: Vlad Zaharia) | 94 |
| Figura 2.58. y Figura 2.59. Fotografía de una pinza de la herramienta de manipulación (Izquierda) y fotografía de un dedo de goma junto a los dedos de aluminio (Derecha). (Fuente: Vlad Zaharia) .. | 95 |
| Figura 2.60. y Figura 2.61. Fotografía del cerrojo visto desde el exterior (Izquierda) y desde el interior (Derecha). (Fuente: Vlad Zaharia) | 95 |

Índice de tablas

| | |
|---|-----|
| Tabla 2.1. Resumen de los requisitos previos del cliente (Fuente: Vlad Zaharia)..... | 31 |
| Tabla 2.2. Velocidades angulares máximas de giro de los ejes del IRB 2600 (Fuente: Manual de usuario IRB 2600) | 40 |
| Tabla 2.3. Pasos y acciones detalladas del ciclo de la celda robotizada (Fuente: Vlad Zaharia) | 58 |
| Tabla 3.1. Resumen de las aportaciones más relevantes del autor al proyecto (Fuente: Vlad Zaharia) | 98 |
| Tabla 4.1. Tabla de los costes materiales de una celda robotizada del presente proyecto. (Fuente: Vlad Zaharia) | 99 |
| Tabla 4.2. Tabla de los gastos de trabajo del proyecto. (Fuente: Vlad Zaharia) | 99 |
| Tabla 4.3. Tabla de las especificaciones del robot IRB 2600 sobre sus niveles de ruido ambiental. (Fuente: Datos técnicos IRB 2600) | 100 |

Prefacio

0.1. Origen del trabajo

El presente proyecto se ha desarrollado durante una estancia de prácticas en la empresa Asea Brown Boveri (ABB) en su planta de Sant Quirze del Vallés, dentro del departamento de ingeniería de la división RM (Robotics and Motion), entre el mes de enero y mayo de 2019.

ABB es una corporación multinacional líder y pionera tecnológica en el sector de la ingeniería industrial. Aunque la empresa tiene origen sueco, como se puede apreciar en sus valores y metodologías de trabajo, su sede se encuentra en Zürich (Suiza), y tiene presencia internacional en más de cien países contando además con más de 135 mil trabajadores. Actualmente los mayores negocios de la compañía son la generación de energía eléctrica y la automatización industrial, sin dejar de banda que se sitúa como la segunda empresa a nivel mundial en venta de robots industriales. La empresa colabora con clientes de compañías de servicios básicos, industrias, medios de transporte e infraestructuras dando un especial enfoque a la digitalización industrial y la nueva industria 4.0. En cuanto a las celdas robotizadas, ABB dispone de productos estandarizados a nivel internacional, teniendo así numerosas soluciones para celdas robotizadas ya prefabricadas. Estas celdas prefabricadas están diseñadas para distintas aplicaciones como la manipulación, la soldadura, la pintura o el escaneo tridimensional de piezas. En el caso de este proyecto se han solicitado ciertas características que no se podrían suplir con una celda prefabricada, por esta razón se ha llevado a cabo un trabajo de diseño de la solución personalizada.

La empresa cliente de este proyecto es una compañía francesa que se posiciona entre los mayores fabricantes de piezas de interior del automóvil, siendo así considerada una empresa “Tier 1” en el sector de la automoción. En este caso la compañía cliente primeramente especifica las características deseadas para la celda robotizada, con el fin de cumplir su necesidad. Y posteriormente, el equipo de proyectos del departamento de ingeniería de ABB les presenta varias soluciones ofertadas, dando así opción a elegir la solución que consideren más adecuada a sus requisitos y presupuesto.

0.2. Motivación

Siendo un alumno de la Ingeniería Electrónica Industrial y Automática de la Universidad Politécnica de Cataluña, y que además siempre ha sentido pasión e interés hacia el mundo de la robótica, el autor de este proyecto tuvo la gran suerte de poder colaborar en un proyecto de esta magnitud y características.

Al formar parte del equipo de proyectos de ingeniería de la división de Robotics and Motion en la planta de ABB situada en Sant Quirze del Vallés, fue destinado a este proyecto cumpliendo las labores necesarias para la elaboración y el desarrollo de este.

La motivación que ha llevado a realizar este proyecto viene dada por los resultados y el proceso de elaboración del proyecto. Se trata de un proyecto muy llamativo a la par que complejo. Adicionalmente quiso poner en práctica los conocimientos adquiridos sobre robótica y visión artificial, al mismo tiempo que se presentaba una gran oportunidad para aprender nuevos conceptos de la industria de la automoción.

0.3. Requisitos previos

Los requisitos previos necesarios a nivel personal para la elaboración de este trabajo de fin de grado han sido en primer lugar formar parte del equipo de proyectos de ingeniería industrial de ABB Robotics and Motion. Este requisito ha garantizado el acceso a todo el material y medios necesarios para la elaboración de este proyecto. El recurso que fue esencial para la elaboración del trabajo en el proyecto fue en primer lugar un ordenador portátil subministrado por la empresa, junto con todas las licencias virtuales que pudiera necesitar, destacando la licencia completa al programa RobotStudio de ABB. Con este programa es con el que se ha llevado a cabo todo el proceso de simulación de la celda robotizada y la programación completa del controlador del robot.

Además, fueron básicos todos los conocimientos adquiridos en el desarrollo de los anteriores proyectos llevados a cabo en el departamento de proyecto de ABB, así como en los cursos de programación de robots y de simulación 3D mediante Robot Studio a los que asistió el autor de esta memoria. Los conocimientos adquiridos durante los estudios de Ingeniería Electrónica Industrial y Automática han resultado imprescindibles para el correcto desarrollo del proyecto, destacando principalmente los adquiridos en la asignatura Robótica Industrial y Visión por Computadora.

Por último, para llevar el trabajo virtual a la instalación real, fueron necesarios los componentes de la celda mencionados más adelante, destacando el robot, la cámara de visión y sus controladores.

Introducción

En el presente trabajo se describen los pasos a seguir para desarrollar un proyecto de la industria automovilística que integra un robot articulado como parte de su solución. Aunque el proyecto realmente consiste en el desarrollo de cuatro celdas idénticas, en este documento se describe el proceso teniendo en cuenta únicamente una de las celdas, debido a la similitud de estas, con la única diferencia en el modelo de pieza que tratan.

La función principal de la instalación es la de ensamblar clips dentro de una pieza de plástico de mayor tamaño. Para esta operación se utiliza un robot articulado industrial IRB 2600, que accede al interior de una máquina de soldadura de plásticos por ultrasonidos, recoge la pieza plástica, que en este caso es el interior de una puerta de automóvil, y posteriormente la coloca en un utillaje para introducirle los clips plásticos y devolverla a la máquina de soldadura.

Las partes del proyecto que más se enfatizan en la presente memoria son: el estudio previo que se ha realizado para el proyecto, la creación de la estación virtual y la simulación 3D, la programación del controlador del robot y la programación del sistema de visión artificial. También se da mucha importancia a la gestión y toma de decisiones en el proyecto además de las modificaciones y la evolución que este ha sufrido.

Con el motivo de respetar el acuerdo de confidencialidad existente, se ha obviado la información sensible sobre la compañía cliente y el proyecto, como por ejemplo cualquier información relacionada con la máquina de soldadura por ultrasonidos. Así mismo, se han sustituido los nombres de los modelos de la pieza para que no se pueda identificar el modelo comercial.

0.4. Objetivos del trabajo

El objetivo principal del presente proyecto es el diseño, la completa elaboración e implementación de la celda robotizada para satisfacer las especificaciones del cliente y la normativa vigente. Además de este, los objetivos descritos en el presente documento son: la elaboración de la simulación 3D de la celda robotizada, la programación del controlador del robot y del controlador de la cámara de visión artificial. Se pondrá especial hincapié en los objetivos anteriormente mencionados debido a que coinciden con los objetivos personales del autor de esta memoria. Así mismo, se introducirán conceptos básicos relacionados con la robótica y los distintos elementos del conjunto de la celda.

Se espera que la presente memoria sirva como guía para futuros proyectos de características similares, facilitando y estandarizando así el proceso de desarrollo y fabricación.

0.5. Alcance del trabajo

Con el fin de elaborar una memoria que pueda usarse en futuros proyectos de la compañía ABB, como guía en proyectos similares, se ha redactado el presente documento que describa el proceso de diseño e implementación de una celda robotizada.

Cabe mencionar que el proyecto realmente consiste de la elaboración de cuatro celdas robotizadas idénticas, con la única variación en la pieza que se fabrica. Con el fin de simplificar a facilitar la comprensión del trabajo, este solo embarca la descripción de una de las cuatro celdas, pero es necesario tener en cuenta que se han repetido las mismas tareas de desarrollo cuatro veces.

Se ha considerado conveniente elaborar un marco teórico que permita al lector entender los conceptos básicos de la robótica, siendo estos claves para el correcto entendimiento del escrito y de los programas en él presentes.

Más adelante se detallará el proceso de desarrollo y gestión del proyecto, ya que el autor se ha visto directamente involucrado en este proceso, así como, el proceso de reunión e interacción constante con el cliente, llegando a tomar algunas decisiones relevantes para el proyecto. Además, se describirán los principales elementos que componen la instalación y la evolución que estos han vivido durante el desarrollo del proyecto.

Se explicarán con mayor detalle los apartados del diseño 3D y la elaboración de la simulación, la programación del controlador IRC 5 que gobierna los robots, y también, la programación del controlador CV-X150RP encargado de la visión artificial de la celda. Este detalle se debe a que fueron tareas elaboradas por el autor de esta memoria, de forma íntegra, como es el caso de la programación de la visión artificial, o en colaboración en el caso de la simulación y la programación del controlador IRC 5. La programación del PLC S7-1515F y su sistema de comunicación queda fuera del alcance de este trabajo.

Esta memoria se estructura en tres capítulos. El Capítulo 1: Marco teórico y estudio de la solución, trata de introducir los conocimientos básicos al lector de este documento con el fin de mejorar su entendimiento. En el Capítulo 2: Desarrollo del proyecto, se detalla el proceso seguido para la elaboración del proyecto. A continuación, se encuentra el Capítulo 3: Resultados, en el que se presenta el resultado final y el proceso de aprobación del cliente. Y finalmente encontramos las conclusiones, estudio económico e impacto medioambiental.

Además, este documento viene acompañado de los Anexos en los que se puede encontrar:

Anexo A1: “User Manual”, es el manual de usuario de la instalación, que cuenta con información detallada de todos los elementos y los procedimientos a seguir.

Anexo B1: Resumen de las normativas aplicables al proyecto.

Anexo B2: Análisis de riesgos y seguridad de la instalación. Consultar este anexo para conocer al detalle las seguridades que se aplican a la celda y las normas que estas cumplen.

Anexo C1: Guía rápida de simulación.

Anexo C2: Guía rápida de programación IRC 5.

Anexo C3: Guía rápida de programación Visión Artificial KEYENCE

Anexo D1: “Communication Interface”, en el que aparece la interfaz de comunicación de la máquina de soldadura con el PLC y el controlador del robot.

Anexo F1: “Layout”, plano de la distribución de los elementos en la celda

Anexo F2: Herramienta de clipaje, en el que se observa un plano de la herramienta de clipaje con sus componentes.

CAPÍTULO I

Marco teórico y estudio de la solución

En este capítulo se aclararán conceptos básicos relacionados con la robótica industrial y los sistemas por los que se rige, además de una presentación de los entornos de programación de los controladores IRC5 de robots de la marca ABB y de los controladores CV-X150RP de las cámaras de visión de la marca KEYENCE. Así mismo, se planteará la solución presentada en este proyecto y como han evolucionado los conceptos durante el desarrollo de este.

1.1. Antecedentes

Para llevar a cabo el proyecto de forma satisfactoria, es necesario entender algunos conceptos relacionados con la robótica y conocer el contexto en el que se sitúa, además de una base histórica que facilita el entendimiento de algunos conceptos y su origen.

1.1.1. Robótica industrial

Según la norma ISO 8373:2012^[1], *“Un robot es un manipulador multifuncional, reprogramable y automático con tres o más grados de libertad y que tanto puede ser fijo o móvil para uso en aplicaciones de automatización industrial”*. La robótica es considerada el principal motor de competitividad y flexibilidad en las industrias de fabricación a gran escala y es un sinónimo de progreso y desarrollo tecnológico, aportando una mejora constante en las prestaciones y aumentando día tras día sus campos de aplicación. A pesar de contar con una gran experiencia que supera las tres décadas, no para de crecer e innovar, posicionándose así entre los sectores con mayores avances tecnológicos. Con el tiempo se ha conseguido mejorar la velocidad de estos, mayor fiabilidad, mayor precisión y aumentar su capacidad de carga, a pesar de la continua disminución de tamaño y peso de los robots.

En los últimos años, se han desarrollado mejoras en el rendimiento y funcionamiento de los robots. Claros ejemplos son la mejora de las comunicaciones, la posibilidad de cooperación y sincronización entre varios robots, además de la facilidad programarlos. Los robots actualmente cuentan con consolas ligeras de programación, que presentan paneles táctiles y en color que permiten su programación de forma muy intuitiva. En los controladores más innovadores estas consolas han sido sustituidas por tabletas táctiles o hasta permiten el control por teléfono móvil. La tecnología actual permite combinar las funcionalidades de los robots industriales con útiles, herramientas, sensores y dispositivos de control, dando así la posibilidad de intervenir en gran número de los procesos de fabricación.



Figura 1.1. Imagen representativa de la aplicación de robots industriales en el sector automovilístico (Fuente: The Robot Report)^[18]

En cuanto al impacto económico que significa el uso de robots industriales, se observa que, en los países más desarrollados, las inversiones en tecnologías robóticas han crecido de forma significativa, superando así a otros sectores tecnológicos. Hay que destacar que en la actualidad la mayoría de los robots están instalados en la industria del automóvil siendo la manipulación y la soldadura las aplicaciones más demandadas. Aunque más de un tercio de los robots industriales se encuentran en Japón, los líderes en la fabricación de robots son fabricantes europeos (ABB, Kuka, Comau, Stäubli).

1.2. Conceptos básicos de los sistemas robóticos

A continuación, se describirán los conceptos básicos necesarios para comprender el funcionamiento del programa y el sistema del robot. Los conceptos descritos a continuación son aplicables a la mayoría de los robots industriales comercializados en la actualidad, aun así, hace falta mencionar que describen especialmente los sistemas de los robots de la compañía ABB, por lo tanto, podría haber alguna variación en los sistemas robóticos de otras marcas comerciales.

1.2.1. Sistema de coordenadas

Los sistemas de coordenadas definen, mediante la orientación de ejes, un espacio, que se origina en un punto fijo llamado origen de coordenadas. Todos los puntos y las posiciones que procesan los robots se sitúan en el espacio referenciados a los ejes de los sistemas de coordenadas. Los robots pueden utilizar varios sistemas de coordenadas dependiendo del tipo de programación de movimiento que se desee ejecutar.

A continuación, se presentan los sistemas de coordenadas más comunes en el uso de los robots industriales:

- El sistema de coordenadas de la base del robot: Este sistema permite mover el robot entre las distintas posiciones de una manera sencilla, ya que una vez el usuario es familiarizado con la orientación de la base del robot, se vuelve un sistema intuitivo.
- El sistema de coordenadas del objeto de trabajo: Está referenciado a la pieza u objeto de trabajo y suele ser el sistema que presenta mejor comodidad para programar el robot, ya que, si un objeto de trabajo es móvil o se modifica su posición, solo será necesario volver a definir el objeto de trabajo para que el robot pueda recalcular los puntos previamente guardados.
- El sistema de coordenadas de la herramienta: Sirve para mover el robot en referencia a la herramienta de trabajo. Este método se suele usar cuando se desea trabajar con movimientos cortos alrededor del TCP (Tool Center Point).
- El sistema de coordenadas del mundo: Este sistema de coordenadas es definido por la celda del robot, y todos los demás sistemas de coordenadas se relacionan, de forma directa o indirecta, al sistema de coordenadas del mundo. Presenta gran utilidad al manejar estaciones y celdas con varios robots o robots movidos por ejes externos.
- El sistema de coordenadas del usuario: Se utiliza para representar conjuntos de sistemas de coordenadas, por ejemplo, objetos de trabajo.

Cuando se desee mover el robot, se debe determinar respecto a que sistema de coordenadas de realizar sus movimientos. En caso de reiniciar el controlador del robot se restablecerá automáticamente a la configuración predeterminada después de un reinicio.

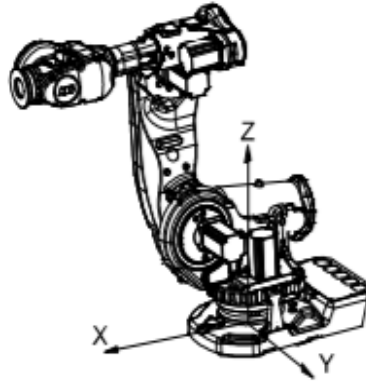


Figura 1.2. Figura que representa el sistema de coordenadas de la base del robot IRB 2600 (Fuente: Manual de instrucciones del IRB 2600)^[14]

1.2.2. Herramienta y “TCP” (Tool Center Point)

El punto central de la herramienta (TCP) es el punto en relación al que se define todo el posicionamiento del robot. Las herramientas se acoplan de forma virtual al sexto eje del robot, por lo tanto, estas serán desplazadas junto con el robot. Este factor servirá de referencia para que el robot pueda recalcular los movimientos y trayectorias teniendo en cuenta el TCP como su extremo. El punto central de la herramienta también constituye el origen del sistema de coordenadas de la herramienta. El sistema de robot puede almacenar y usar varios datos de herramientas, siendo posible tener activa solo una herramienta al mismo tiempo. Las cargas útiles son importantes cuando se trabaja con pinzas, ya que será un factor que el controlador tendrá en cuenta a la hora de establecer las aceleraciones y velocidades en las que realizará el movimiento.

En el presente proyecto se usan tres herramientas distintas, y cada una de ellas cuenta con un número diferente de TCPs debido a la necesidad propia de la herramienta.

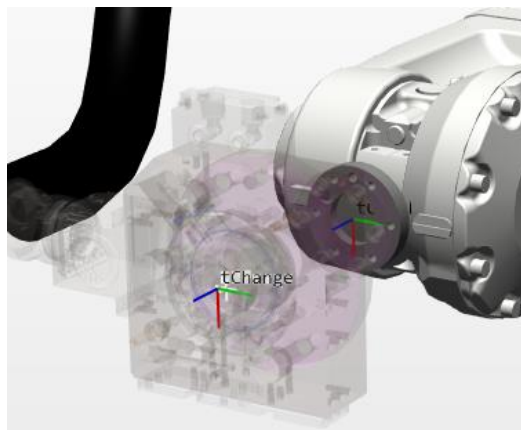


Figura 1.3. Captura representativa del TCP de la herramienta tChange (Fuente: Simulación Vlad Zaharia)

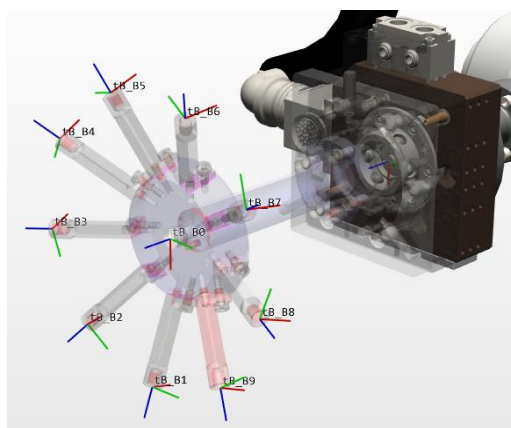


Figura 1.4. Captura representativa de los 10 TCPs de la herramienta tB (Fuente: Simulación Vlad Zaharia)

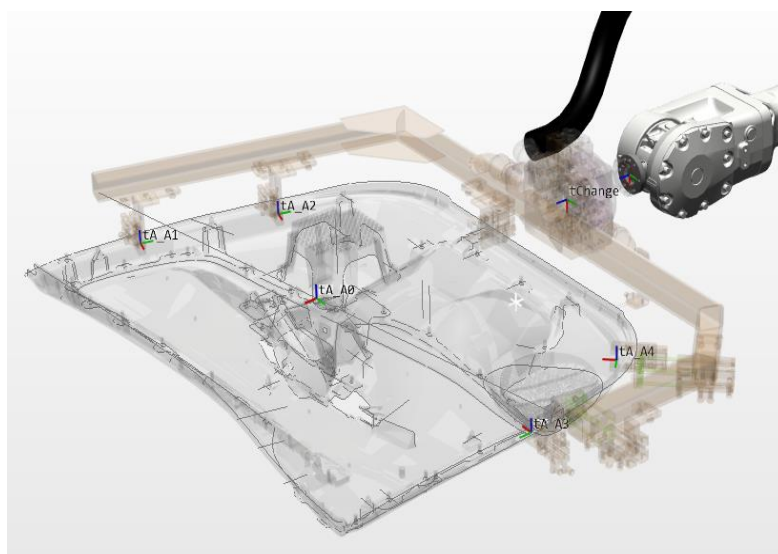


Figura 1.5. Captura representativa de los 5 TCPs de la herramienta tA (Fuente: Simulación Vlad Zaharia)

1.2.3. Objeto de trabajo

Se usan principalmente para simplificar en gran medida la programación cuando es necesario editar programas debido a desplazamientos de trayectorias o procesos que se adhieren a un objeto físico. Los objetos de trabajo a menudo se crean para simplificar el desplazamiento a lo largo de las superficies de un objeto. Como es el caso de las herramientas, puede haber varios objetos de trabajo creados, pero se debe elegir cuál usar en cada movimiento que ejecuta el robot. En caso de que se desee manipular un objeto de trabajo se debe tener en cuenta su peso, ya que es un factor clave para el cálculo de aceleraciones del robot.



Figura 1.6. Imagen del objeto de trabajo de clipaje donde se coloca la pieza a manipular (Fuente: Vlad Zaharia)

Existen varios métodos de declaración de objetos de trabajo, pero el más común es la indicación precisa al robot de tres puntos ortogonales de la superficie del objeto de trabajo.

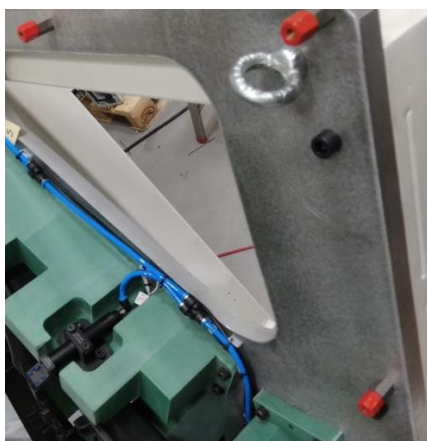


Figura 1.7. y Figura 1.8. Situación de los tres puntos ortogonales que definen un objeto de trabajo (izquierda) y punto metálico usada para obtener mayor precisión en la declaración de objetos de trabajo (derecha) (Fuente: Vlad Zaharia)

1.2.4. Movimiento

Tras detallar los sistemas de referencia de los movimientos de los robots, se procede a explicar los modos de movimiento y las configuraciones que se deben tomar en cuenta.

En el **Modo lineal** de movimiento el sistema desplazará el TCP en referencia al sistema de coordenadas seleccionado.

El **Modo de reorientación** consiste en usar el sistema de coordenada de la herramienta para llevar a cabo una reorientación de esta.

El **Modo de movimiento por ejes** consiste en mover los ejes que forman la unidad mecánica de forma independiente, pudiendo así determinar el ángulo de giro de cada eje.

En las consolas de programación ABB, se puede acceder al menú de movimiento y observar la siguiente información del movimiento del robot. Estas características pueden ser modificadas según los requisitos del usuario o programador:

- **Unidad mecánica:** Muestra que unidad mecánica se encuentra activa y permite cambiarle en caso de que exista más de una unidad mecánica (varios robots o ejes externos, como mesas rotatorias o transportadores).
- **Modo de movimiento:** Selecciona el modo de movimiento en el que se desea desplazar la unidad mecánica seleccionada. En el caso de los ejes externos permite seleccionar su posición relativa.
- **Sistema de coordenadas:** puede cambiar el sistema de coordenadas en base al que se mueve el robot.
- **Herramienta:** Selecciona la herramienta activa para realizar el movimiento. Se suele seleccionar la herramienta como sistema de coordenadas para moverse cuando está cerca de un punto de trabajo o para mover el robot en las direcciones de la herramienta o reorientarlo.
- **Objeto de trabajo:** Selecciona el objeto de trabajo respecto al que se desplaza el robot. Se utiliza para mover la unidad mecánica activa en las direcciones del objeto de trabajo, siendo este el lugar donde trabaja el robot.
- **Carga útil:** Se debe declarar si el robot trae carga o no. Las cargas de las herramientas y de los objetos de manipulación deben ser declarados previamente. Los robots ABB cuentan con una rutina que tiene la función de calcular la carga que sujeta el robot de forma automática.
- **Incremento:** Permite que el robot se mueva en incrementos de distintos tamaños. Suele usarse para conseguir movimientos de precisión, o guardar puntos de forma muy precisa.

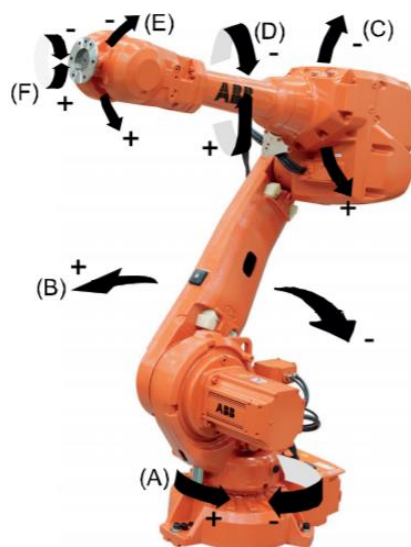


Figura 1.9. Imagen que muestra la situación de los ejes del robot, junto a su nomenclatura y el sentido de giro (Fuente: Manual del robot IRB 2600)^[14]

| Posición | Descripción |
|----------|-------------|
| A | Eje 1 |
| B | Eje 2 |
| C | Eje 3 |
| D | Eje 4 |
| E | Eje 5 |
| F | Eje 6 |

Tabla 1.1. Tabla que muestra la equivalencia de nomenclatura de los ejes del robot (Fuente: Manual del robot IRB 2600)^[14]

1.2.5. RobotStudio

RobotStudio es el software de simulación y programación fuera de línea de ABB, y permite efectuar la programación del robot desde un ordenador de forma remota sin la necesidad interrumpir la producción, como se producía con anterioridad en los controladores robóticos. La evolución de los programas específicos de simulación de robots permite recrear el entorno real de trabajo y la programación de los robots de forma offline. Este factor permite maximizar la eficiencia, el aumento de la productividad y el ahorro económico.

RobotStudio proporciona las herramientas para la programación y optimización de las celdas robotizadas, sin afectar la producción, lo que permite reducir riesgos y disminuir el tiempo de puesta en marcha incrementando la productividad.

RobotStudio ha sido diseñado mediante el VirtualController de ABB, que permite recrear de forma precisa el software real que hace funcionar el robot. Además, permite realizar simulaciones muy realistas, con archivos de configuración y programas de robot idénticos a los utilizados en la instalación final.

En este proyecto se ha desarrollado tanto la simulación como la programación usando un sistema robótico virtual mediante el uso del software mencionado. Esta herramienta, aparte de permitir ver tridimensionalmente los diferentes elementos que forman la estación y comprobar la correcta distribución del espacio, permite asegurar que no se producen colisiones entre los elementos. Posteriormente se ha creado una simulación muy fiel al proceso real que ha permitido guiar el proceso de programación para poder así agilizarlo. Una vez finalizada la simulación esta será directamente introducida al controlador del robot, que permite ejecutar el programa de la simulación con total fidelidad.

1.2.6. Visión artificial (KEYENCE)

La visión artificial o visión por ordenador es la disciplina científica que consiste en que los ordenadores o controladores imiten el proceso de visión humano, con la ayuda de sensores de imagen electrónicos, y la finalidad de obtener información numérica o simbólica que posteriormente pueda ser procesada. El proceso más común seguido en los sistemas de visión artificial es el de adquisición, procesamiento, análisis y posterior comprensión de las imágenes.

La visión artificial se puede usar en numerosos campos de aplicación, presentando un amplio abanico de posibilidades. Algunos ejemplos son: el control de procesos, la detección de eventos, la interacción o inspección automática, o hasta algunas aplicaciones más avanzadas como, la navegación de los vehículos autónomos o robots AVG, o el modelaje 3D de objetos. El procesamiento de imagen con objetivo de determinar si una imagen contiene un objeto, característica o actividad es la tarea más utilizada en el campo de la visión artificial. Con el fin de reconocer los elementos deseados, se deben cumplir unas condiciones adecuadas de iluminación, entorno y posición relativa a la cámara, asegurando así homogeneidad y repetitividad en las capturas.

El controlador CV-X150RP de la marca KEYENCE usado en este proyecto para el control de la cámara, con la finalidad de detección de objetos (clips), dispone de un sistema operativo y software propio, suministrado por el fabricante, con la finalidad de facilitar la experiencia de usuario, siendo esta más intuitiva. Al mismo tiempo, también se distribuye el Terminal Software que permite la conexión remota

al controlador para poder programarlo de forma remota desde un ordenador personal y el Simulation Software que permite trabajar de forma offline sin interrumpir el proceso de trabajo de la instalación. En el presente proyecto no ha sido necesario el uso del Simulation Software ya que la programación se ha realizado de forma directa en los controladores CV-X150RP.

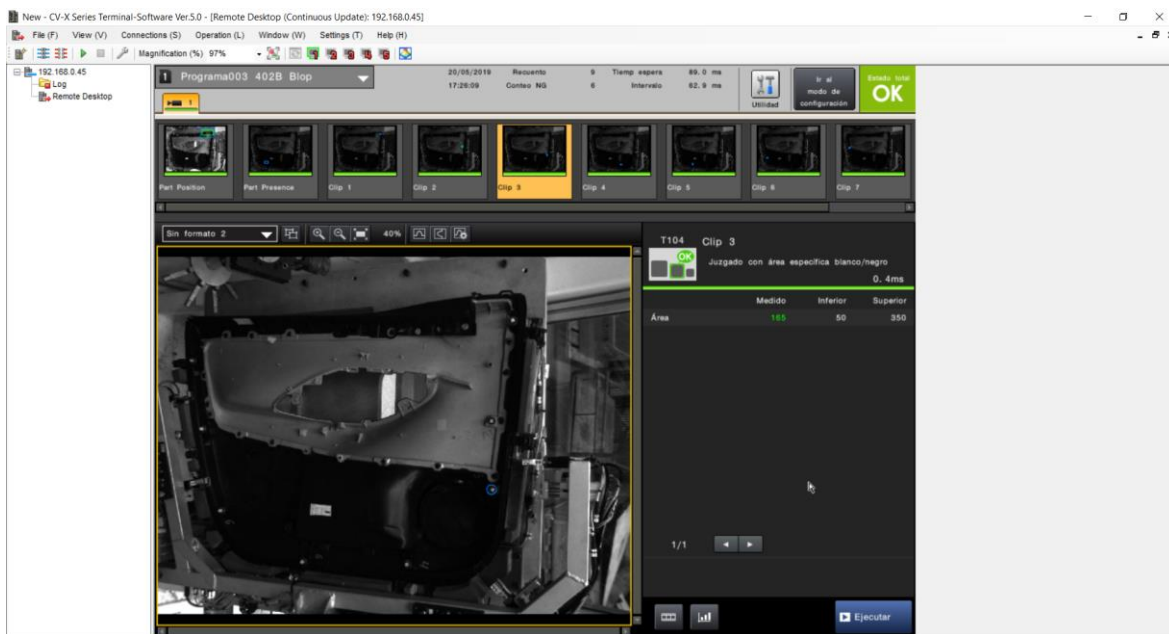


Figura 1.10. Ejemplo de captura de la interfaz del Terminal Software de KEYENCE (Fuente: Vlad Zaharia)

CAPÍTULO II

Desarrollo del proyecto

En el siguiente capítulo se describe exhaustivamente el proceso seguido en el desarrollo del proyecto, dando enfoque a la gestión del proyecto y los procedimientos seguidos para la elaboración de las tareas necesarias. Se detallan especialmente los procesos de simulación, y programación tanto del controlador IRC5 como del CV-X150RP.

2.1. Requisitos previos y estudio de alternativas

Teniendo en cuenta de que el cliente ya dispone de unas celdas de similares características en una de sus plantas, pudo dar detalles muy específicos de como deseaba que fuera la celda. Una de las características principales, y la que más restricciones produjo a lo largo del desarrollo del proyecto ha sido el tamaño de la celda robotizada. El cliente necesitaba una solución estrecha (2020 mm) para minimizar el espacio entre las celdas, disminuyendo así al máximo el tiempo de desplazamiento del operario entre celda y celda, llegando así a reducir en uno el número de empleados necesario para la elaboración de la pieza final.

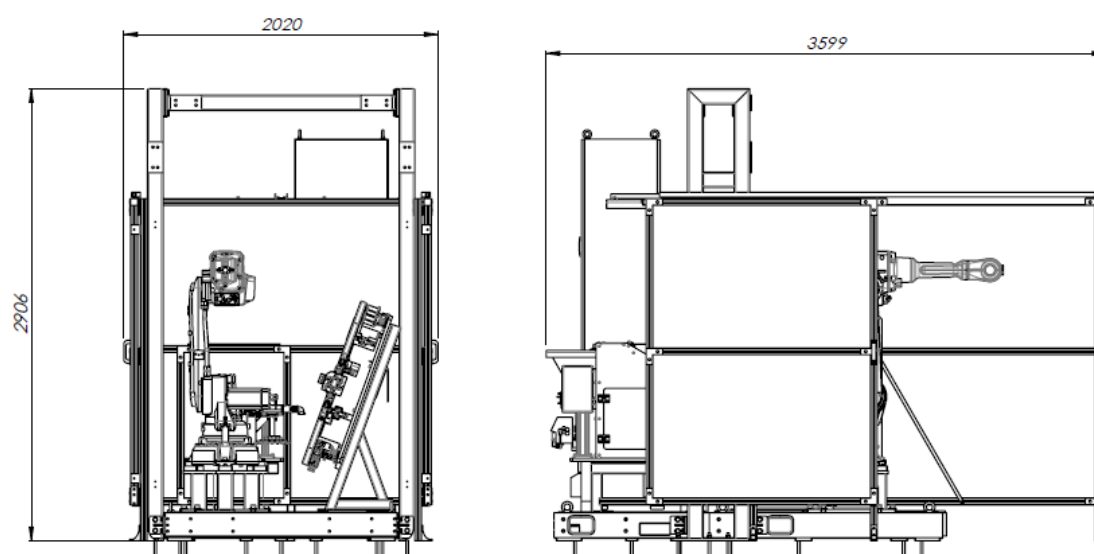


Figura 2.1. y Figura 2.2. Medidas de la celda final desde la vista frontal (Izquierda) y lateral (Derecha) (Fuente: Anexo F1: “*LayOut*”)

La función descrita por el cliente, que debe cumplir la celda es la de ensamblar clips dentro de una pieza de plástico del interior de las puertas de un modelo determinado de automóvil. Para esta el robot primeramente debe acceder al interior de una máquina de soldadura de plásticos por ultrasonidos, recoger la pieza plástica, y posteriormente colocarla en el utillaje de clipaje para introducirle los clips plásticos y devolverla a la máquina de soldadura, finalizando así el ciclo.

La única función del operario en esta instalación es retirar la pieza ya finalizada y sustituirla por una nueva. Esta acción la realiza mientras el robot este clipando la pieza anteriormente introducida, de este modo cuando el robot vuelva al interior de la máquina de soldadura ya dispondrá de una pieza nueva a la que manipular.

También fueron detalladas ciertas funcionalidades extras que debe realizar la instalación. Por ejemplo, esta debe ser capaz de inspeccionar si los clips han sido colocados correctamente, y en caso de que no sea así, en el siguiente ciclo debe realizar un proceso de limpieza de herramienta de clipaje de forma automática, evitando la intervención del operario de planta. Todos estos requisitos vienen acompañados de requisitos de seguridad y normativa que debe cumplir la máquina para su validación. Estas normas están descritas en los Anexos B1 y B2 de este documento.

Tanto el modelo del robot (IRB 2600), como el controlador (CV-X150RP) y el sensor (CV-200M) usados para la inspección fueron requisitos del cliente. Este hecho se debe a que ya se han realizado previamente proyectos de características similares y el cliente ya está familiarizado con tales productos y los costes que estos suponen. Por lo tanto, se procedió a elaborar una solución que incorporara estos componentes y que además minimizara el coste de fabricación de la celda y facilitara reducir el tiempo de ciclo de fabricación.

A continuación, aparece una tabla que resume los requisitos previos más importantes:

| Número | Requisito |
|--------|--|
| 1 | Tamaño de la celda: 2020 mm de ancho y 3600 mm de largo. |
| 2 | El uso de un único robot articulado del modelo IRB 2600. |
| 3 | El uso de la cámara CV-200M y el controlador CV-X150RP para realizar la función de detección de clips. |
| 4 | Que se diseñe una interfaz de comunicación ente el PLC y la máquina de soldadura. |
| 5 | Que sea el PLC el que gobierne las señales y las seguridades de la celda. |
| 6 | Que haya suficiente espacio entre la plataforma del robot y la máquina de soldadura con el fin de introducir el carro de mantenimiento en el interior de la celda. |
| 7 | Que la celda realice funcionalidades secundarias: limpieza y alimentación de los clips. |

Tabla 2.1. Resumen de los requisitos previos del cliente (Fuente: Vlad Zaharia)

Inicialmente se planteó situar el utillaje de clipaje entre el robot y la máquina de soldadura en una situación horizontal, dejando así el espacio trasero de la celda para los procesos de alimentación de

clips de la pinza y la limpieza, y también reduciendo las distancias de movimiento del robot. Esta solución fue rápidamente descartada debido a la poca practicidad que aparecía a la hora de introducir el carro de mantenimiento dentro de la celda.

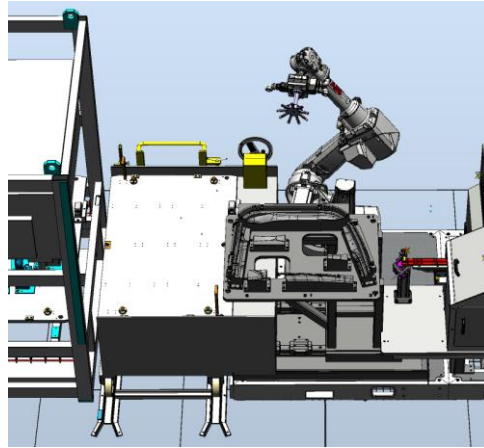


Figura 2.3. Captura del modelo 3D de la primera solución presentada (Fuente: Vlad Zaharia)

Otra de las características del diseño inicial fue colocar la cámara con la ayuda de un soporte en la parte superior de la celda y realizar las capturas de las imágenes mientras el robot está manipulando la puerta.

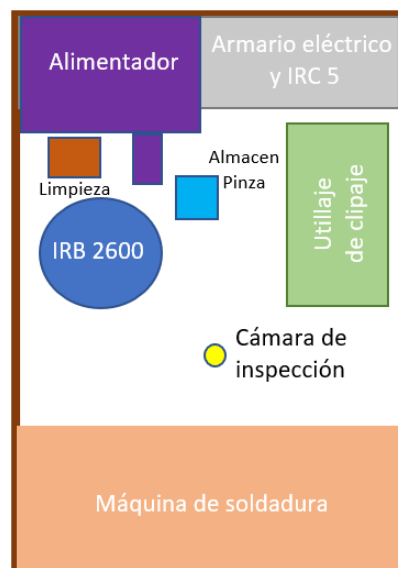


Figura 2.4. Croquis de la solución inicial (Fuente: Vlad Zaharia)

Como se verá posteriormente esta solución ha sido modificada con el fin de mejorar el funcionamiento de la máquina y permitir un mejor acceso a su interior.

2.2. Metodología de desarrollo de proyecto

Al recibir las especificaciones del cliente lo primero que sucede es que se reúne el equipo de ingenieros destinados a este proyecto y se discute una posible solución a la necesidad del cliente. Al ser un diseño nuevo, aunque con algunos precedentes, se deben elegir los elementos que componen la celda. Debido a que el departamento de proyectos de ingeniería de ABB ya había diseñado celdas de clipaje de características similares, se ha optado por usar soluciones y componentes previamente usados en otros proyectos del mismo cliente.

Para plasmar de forma más gráfica la solución, se desarrolla una estación 3D con la ayuda del programa RobotStudio, que engloba los modelos 3D de los elementos y productos suministrados por los proveedores. La finalidad es asegurar que se cumple con las especificaciones de tamaño de la celda, y además proporcionar una visión más cómoda de la solución presentada. Para esta tarea se ha destinado un Ingeniero mecánico experto en diseños de celdas y simulaciones, y además este ha recibido soporte por parte del autor de este document, en algunas tareas de estudio y colocación de los modelos 3D en la estación.

Tras determinar de forma concreta cuáles son los elementos que compondrán la celda se procede a acordar la oferta con el cliente y en pocos días se inicia el desarrollo del proyecto. Cuando el diseño es aceptado por el cliente, se realiza una lista de los diferentes elementos que se deben comprar y se crean los respectivos pedidos. Además, se externaliza el diseño y la fabricación de aquellos que no están en el catálogo de productos de ABB como es el caso de las herramientas de manipulación y clipaje, o el utillaje. Una vez los elementos y componentes comandados llegan a la planta de Sant Quirze del Vallés se inicia el montaje de la celda y la instalación de las seguridades para garantizar la seguridad de los técnicos que trabajan en el entorno. A medida que estos elementos son montados en la celda se valida el correcto funcionamiento de todos ellos.

El montaje de la celda se lleva a cabo por una empresa subcontratada especializada en montajes mecánicos y eléctricos. En la creación de esta instalación se han destinados dos técnicos mecánicos y dos eléctricos. Esta misma empresa es la encargada de realizar los planos eléctricos y neumáticos de la celda, que se adjuntarán a la documentación final del proyecto. Además, se ha tomado la decisión de subcontratar también un programador de PLC y un programador de robots de otra empresa especializada en sistemas de gestión, automatización y robótica. Por parte de ABB se ha destinado otro programador de robots, que en este caso ha sido el autor de la presente memoria, con la finalidad de transmitir la experiencia previa en proyectos con características similares y garantizar que se siguen los estándares de programación de robots establecidos recientemente. Antes de la incorporación en este proyecto, éste elaboró un manual de programación de robots junto con la ayuda de un programador experto, con el objetivo de que los proyectos que se realicen en el futuro sigan las pautas marcadas.

A la hora de hacer el diseño de la celda se recomienda hacer una identificación de todos los riesgos que se pueden generar durante todas las fases de trabajo, incluyendo el montaje, la programación, el mantenimiento y el uso normal. Después de elaborar una solución concreta y tener el modelo 3D de la estación, se inicia el cálculo del riesgo de la celda, donde se calculan las probabilidades de que se produzcan situaciones de riesgo y ponderando el valor con el daño que puede causar. Estos cálculos se vuelven a realizar cuando el proyecto queda finalizado y se hace una comparación de las soluciones implementadas para reducir los riesgos.

Siguiendo las recomendaciones que aparecen en la *“Guía para elaboración de proyectos”*^[17] emitida por la Viceconsejería de formación profesional y aprendizaje permanente del Gobierno Vasco, se ha podido realizar de forma satisfactoria la monitorización permanente del proyecto recibiendo constante realimentación del cliente de la evolución del proyecto y la toma de decisiones en este. Se han llevado a cabo reuniones quincenales y desde la emisión de la orden de montaje de los componentes se han recibido 5 visitas del propio cliente en la planta. Durante estas visitas principalmente se ha supervisado el estado de la instalación y se han solucionado las dudas o necesidades que pudieran surgir. El autor de este documento ha sido uno de los encargados de recibir al cliente durante sus visitas y de reportar el avance del proyecto, además de asistir a las reuniones quincenales.

Los objetivos de las reuniones quincenales eran: en primer lugar, aprobar el trabajo realizado desde la anterior reunión, analizar posibles causas de los retrasos producidos, tomar las medidas oportunas para evitar estos retrasos y garantizar el cumplimiento de la fecha de entrega, determinar de forma detallada lo que se deberá realizar hasta la fecha de la siguiente reunión acordada.

La existencia de las reuniones y visitas periódicas no quita el hecho de que durante todo el proceso ha existido una comunicación continua, tanto por correo electrónico como por teléfono, entre el propio equipo de ABB como con el cliente. Esta constante comunicación ha sido uno de los factores claves del correcto desarrollo del proyecto.

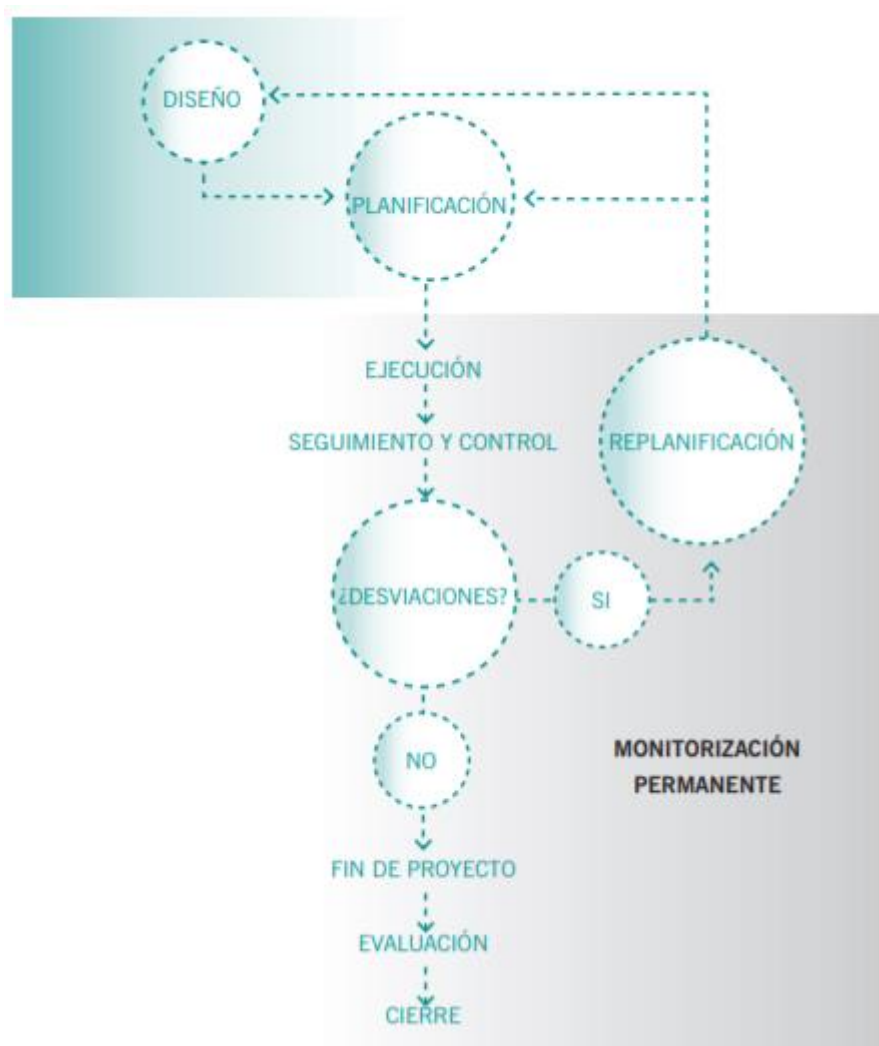


Figura 2.5. Diagrama de flujo del desarrollo del proyecto. (Fuente: Guía para elaboración de proyectos)^[17]

Las funciones principales que cumplió el autor de este proyecto en el desarrollo de este fueron la ayuda en el diseño 3D y la elaboración de la simulación, para poder observar de forma realista si la solución presentada es factible y cumple con los requisitos necesarios. Además, colaboró con otro compañero en llevar a cabo la programación del controlador IRC 5 que gobierna los robots, y también, de forma individual llevó a cabo la programación del controlador CV-X150RP encargado de la visión artificial de la celda. Así mismo, tuvo presencia de forma constante en todas las reuniones llevadas a cabo con el cliente, llegando a influenciar en algunas tomas de decisión relevantes en el proyecto.

A la hora de entregar la celda, el Project Manager redacta una declaración UE (Anexo B1) de conformidad en la que asegura que se cumplen las principales directivas europeas que debe cumplir.

2.2.1. Planificación

La planificación del proyecto es un factor clave para que este se desarrolle de forma satisfactoria. Tras presentarle una solución inicial al cliente, también se establecen los tiempos de cada paso del proyecto. Esta clase de proyecto se suelen dividir por semanas. El cumplimiento de los tiempos establecidos se supervisa en las reuniones quincenales con el cliente, además de en las comunicaciones telefónicas o electrónicas.

A continuación, se representa un diagrama de Gantt de la evolución del proyecto en todas sus fases y las visitas de seguimiento del cliente.

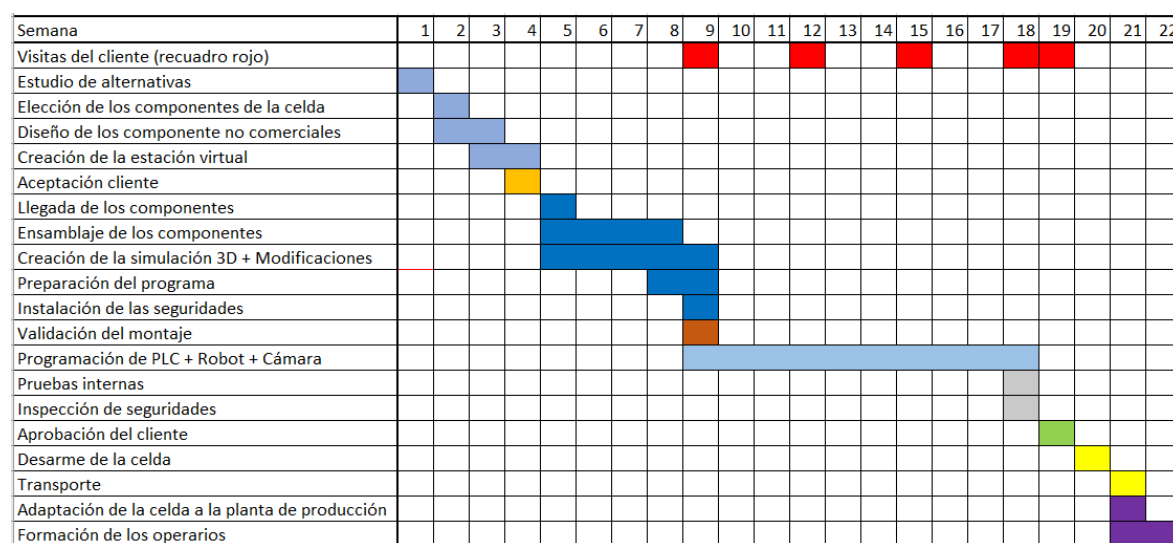


Figura 2.6. Diagrama de Gantt del desarrollo del proyecto. (Fuente: Guía para elaboración de proyectos)

Como se aprecia en el diagrama de Gantt, el tiempo total necesario para el desarrollo del proyecto ha sido de 22 semanas.

2.3. La celda y sus componentes

El objetivo de la instalación, con robot IRB 2600, es el montaje de clips en la pieza de plástico del interior de una puerta de coche. Se han creado cuatro celdas iguales, una para cada puerta del coche: FR (Frontal Left) FR (Frontal Right) RL (Rear Left) RR (Rear Right), aunque en la presente memoria solo se describe el desarrollo de una de las celdas, con la razón de que las cuatro celdas son idénticas, con la única diferencia en la pieza a ensamblar. Las piezas ubicadas en las puertas delanteras pueden variar entre dos modelos. Para conservar la confidencialidad estos dos modelos serán nombrados **Modelo 1** y **Modelo 2** en el presente trabajo. La instalación no está diseñada para otros fines diferentes a los que se explican en este documento.

Como requisito del cliente y formando parte del proceso habitual de los proyectos similares, además de los componentes de la celda se hace entrega de toda documentación que pueda ser necesaria tras la puesta en marcha de la instalación. Debido a la localización final de esta instalación será en Francia, la documentación se elabora en inglés y posteriormente es traducida al francés. Uno de estos documentos es el manual de instrucciones, que se encuentra adjunto en el anexo A1 de la presente memoria. En este se indica de forma detallada el uso correcto y seguro de la celda que se describe en detalle en el anexo B2. Deben respetarse las instrucciones de seguridad marcadas en el manual y las instrucciones generales para el funcionamiento de la maquinaria. También se deben tener en cuenta las medidas de prevención de accidentes y las normas de seguridad generales indicadas en el lugar de trabajo y el trabajo asociado con este equipo. Antes de realizar cualquier trabajo en la máquina, el trabajador debe haber leído el manual de instrucciones y sus manuales asociados, especialmente el capítulo sobre instrucciones de seguridad básicas.

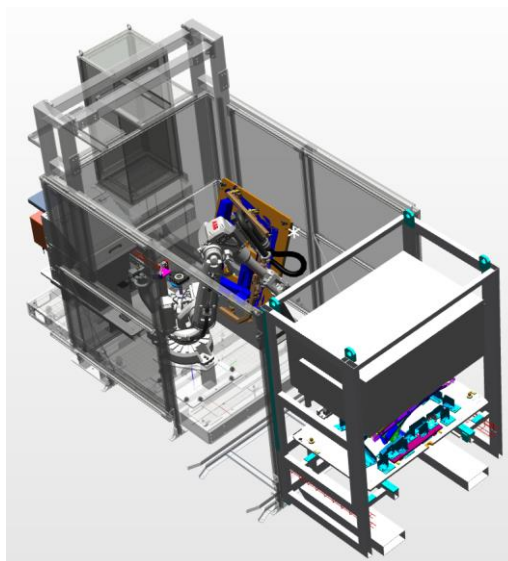


Figura 2.7. La estación 3D de la instalación (Fuente: Vlad Zaharia)

2.3.1. Componentes de la celda

A continuación, se muestra una lista resumida de los principales elementos de la celda. Esta lista fue inicialmente ofertada al cliente con los componentes que formaban la solución y ha sido modificada durante el transcurso del desarrollo del proyecto con el fin de mejorar la instalación.

- 1 x Robot modelo IRB 2600 20/1.65.
- 1 x Utillaje (En las celdas de puerta frontal en el mismo utillaje encajan tanto el Modelo 1 como el Modelo 2) con un sistema de agarre por cilindros abrazadores.
- Estructura de celda metálica: MLTA 1 (Componente comercial).
- 1 x Alimentador de clips de plástico con mono dispensador.
- 1 x Plataforma para el robot
- 1 x plataforma para armarios y alimentador, localizada en la parte trasera
- 1 x Vallado de seguridad
- 2 x Puertas de acceso
- 2 x Teclados de acceso
- 1 x Baliza luminosa.
- 1 x PLC Siemens S7-1515F con safety
- 1 x Módulo de INPUT/OUTPUT
- 1 x HMI Siemens Confort Panel TP900 con Sistema Up&Down.
- 1 x Cámara KEYENCE de 2 Mpx B/N (modelo CV-200M)
- 1 x Controlador CV-X150RP
- 1 x Lente CA-LH8
- 1 x Módulo de iluminación
- Cables eléctricos y canal de cableado para conectar los equipos aquí mencionados.
- 1 x Paquete eléctrico para el robot IRB 2600 (DressPack)
- 1 x Herramienta con 9 dedos para el proceso de clipaje
- 1 x Herramienta con 4 pinzas para manipular la pieza
- 1 x Almacén para la herramienta de clipaje
- 1 x Almacén para la herramienta de manipulación situado en el utillaje
- 1 x Sistema de limpieza de clips.

2.4. Elementos principales de la máquina

Para poder entender la dinámica de funcionamiento de la celda y las distintas operaciones que esta puede llevar a cabo, es necesario describir los distintos elementos principales que forman la instalación. En el presente documento no se mencionan detalles sobre la máquina de soldadura presente en la celda robotizada, con el motivo de preservar la confidencialidad con el cliente.

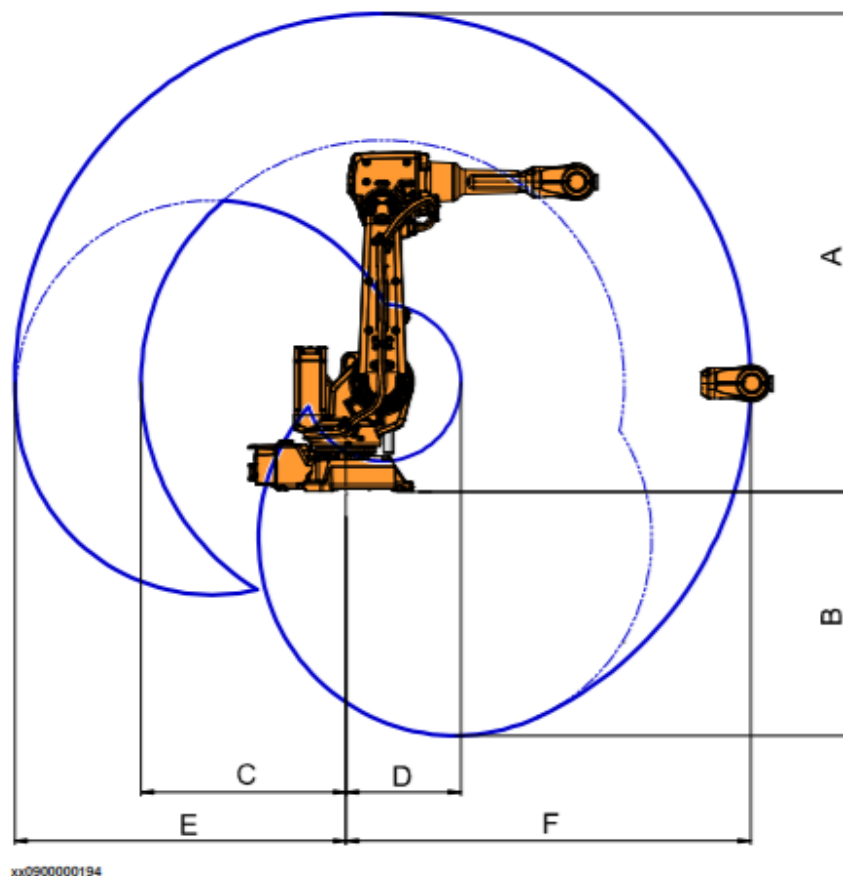
2.4.1. Robot IRB 2600-20/1.65

El modelo seleccionado por el cliente para esta instalación ha sido el IRB 2600-20/1.65. El criterio seguido para esta elección ha sido en primer lugar el peso máximo que puede manipular el robot, que en este caso es de 20 kg, y el tamaño de este, ya que el espacio de maniobra está muy limitado por el reducido tamaño de la celda.

Los robots articulados de la marca ABB se caracterizan por tener resoluciones muy pequeñas en todos los ejes, ofreciendo así una repetibilidad muy elevada en sus movimientos. Esto significa que todas las veces que el robot ejecute una trayectoria, esta únicamente podrá variar del recorrido teórico con un margen menor a 0,13 mm, en el caso del modelo IRB 2600-20/1.65.



Figura 2.8. Imagen del robot IRB 2600-20/1.65 (Fuente: Manual de usuario IRB 2600)^[14]



| Variant | Pos. A | Pos. B | Pos. C | Pos. D | Pos. E | Pos. F |
|--------------------|---------|--------|--------|--------|---------|---------|
| IRB 2600 - 20/1.65 | 1948 mm | 993 mm | 837 mm | 469 mm | 1353 mm | 1653 mm |

Figura 2.9. Representación del área de trabajo del IRB 2600-20/1.65 (Fuente: Manual de usuario IRB 2600)^[14]

Otro factor importante a la hora de elegir el modelo de robot es la velocidad angular máxima de giro de los ejes. Cuanto mayor sea el tamaño del robot, por norma menor es la velocidad de giro de sus ejes, debido al aumento de carga e inercia que representa.

| Robot type | Axis 1 | Axis 2 | Axis 3 | Axis 4 | Axis 5 | Axis 6 |
|--------------------|---------|---------|---------|---------|---------|---------|
| IRB 2600 - 20/1.65 | 175 °/s | 175 °/s | 175 °/s | 360 °/s | 360 °/s | 500 °/s |

Tabla 2.2. Velocidades angulares máximas de giro de los ejes del IRB 2600 (Fuente: Manual de usuario IRB 2600)^[14]

Otra consideración a tener en cuenta es que el rango de temperaturas de trabajo del manipulador está entre +5 y +45°C. En los casos en los que la temperatura sea inferior a los 10°C, se recomienda arrancar con una fase de calentamiento de la maquinaria. Esto se debe a que a baja temperatura hay riesgo de que aumente la viscosidad del aceite y la grasa de los motores, reduciendo así el rendimiento del robot.

Por último, el grado de protección que se le aplica al modelo IRB 2600 es el IP67, que según la normativa CEI 60529 significa que está diseñado para impedir la entrada de polvo bajo cualquier circunstancia en el interior de los mecanismos y la entrada de agua en caso de inmersión completa a 1 metro durante 30 minutos. Este factor no afecta en el caso de la presente instalación ya que no tendrá que funcionar bajo condiciones de sumersión o de elevada cantidad de polvo.

2.4.2. Controlador IRC 5

Toda la gama de robots articulados de ABB actualmente son controlados por el controlador IRC 5, siendo este la última generación de estos controladores. El controlador IRC5 contiene los elementos electrónicos necesarios para controlar el manipulador, los ejes externos y los equipos periféricos. El controlador principal es un Intel Celeron 1.2 GHz con 64 MB de DRAM, y tiene suficiente capacidad para gobernar hasta 36 ejes, pudiendo controlar como máximo 3 robots y varios ejes externos, y realizar operaciones simultáneas por varios canales Ethernet.^[12]

La cabina dispone de una puerta frontal que permite acceder al interior y que puede sellarse, ofreciendo así una protección IP54. Dentro del armario se introducen de manera compacta todos los elementos necesarios para el funcionamiento del robot. Estos elementos son los siguientes:

- El módulo de accionamiento de los motores del robot, que permite añadir hasta 3 ejes externos adicionales
- El módulo de control que incluye el ordenador principal
- Las interfaces de comunicación de varios buses de campo
- Los puertos Ethernet

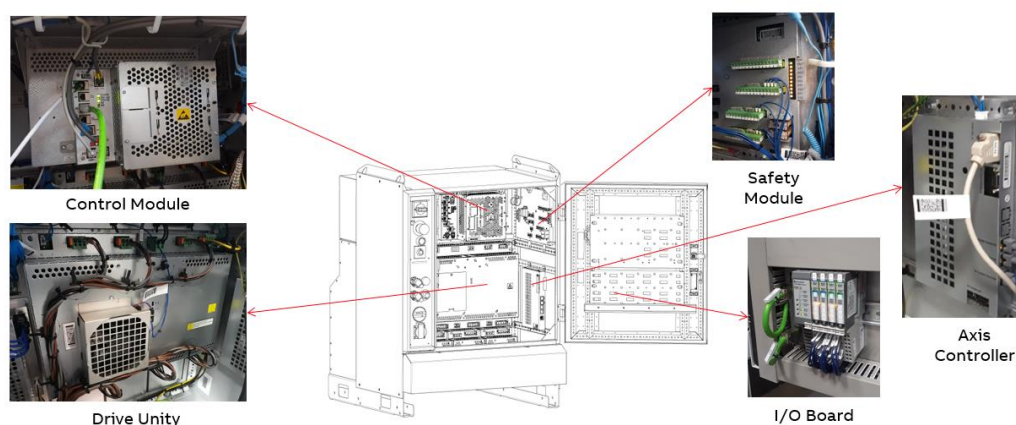


Figura 2.10. Distribución de los distintos módulos y unidades que forman el IRC 5 (Fuente: Vlad Zaharia)

Las diferentes unidades están diseñadas para ser accesibles sin tener que desconectar ni sacar ningún otro elemento. Además, la cabina dispone de conectores de entrada para recibir la corriente trifásica y de salida para llevar la potencia y el control al robot u otras unidades mecánicas.

En su parte lateral izquierda tiene un panel de control, formado por:

- El interruptor de potencia
- Un pulsador de emergencia
- El botón para subir los motores
- Un selector de modo, que permite seleccionar entre el modo Automático, el modo Manual a velocidad reducida, y el modo Manual al 100 %
- Un puerto para conectar la unidad de programación
- Un puerto de servicio para conectarse con el ordenador

En referencia a los modos de operación, el modo manual es el modo seguro de movimiento del robot ya que está destinado a que el robot sólo se pueda mover mientras esté presionado el pulsador de habilitación de la unidad de programación. En este caso la velocidad es reducida y segura tanto para el operario como por la unidad mecánica. Este modo de funcionamiento se utiliza principalmente en la fase de programación o cuando se produzca algún fallo que interrumpa la producción habitual.

El modo automático es el modo habitual de funcionamiento del robot durante la producción. En este modo la velocidad que podría alcanzar el robot sería de 5000 mm/s. Esto implica que en modo automático nadie puede estar en el área de trabajo del robot ya que estaría en riesgo. Por esta razón se incorporan ciertas medidas de seguridad para evitar que el operario y las partes móviles no colisionen. El armario dispone de una tarjeta de entradas y salidas específica para seguridad que se conecta de forma directa a los contactores del armario eléctrico permitiendo detener la instalación rápidamente en el momento en que aparezca un error en la ejecución del controlador.

2.4.3. Consola de programación (FlexPendant)

El controlador IRC 5 incluye también una unidad de programación llamada FlexPendant, que es un dispositivo que se utiliza para la gestión y movimiento del sistema robótico. Esta unidad de programación sirve como interfaz entre el operario y el ordenador del controlador. En el modo de funcionamiento automático esta consola permite comprobar valores de señales y variables, ejecutar el programa y sirve como interfaz para recibir o introducir información al robot. Si el controlador está en modo manual, la consola permite además modificar los programas, mover el manipulador mediante movimientos lineales, reorientaciones o movimientos de ejes, restablecer copias de seguridad, y demás funcionalidades.

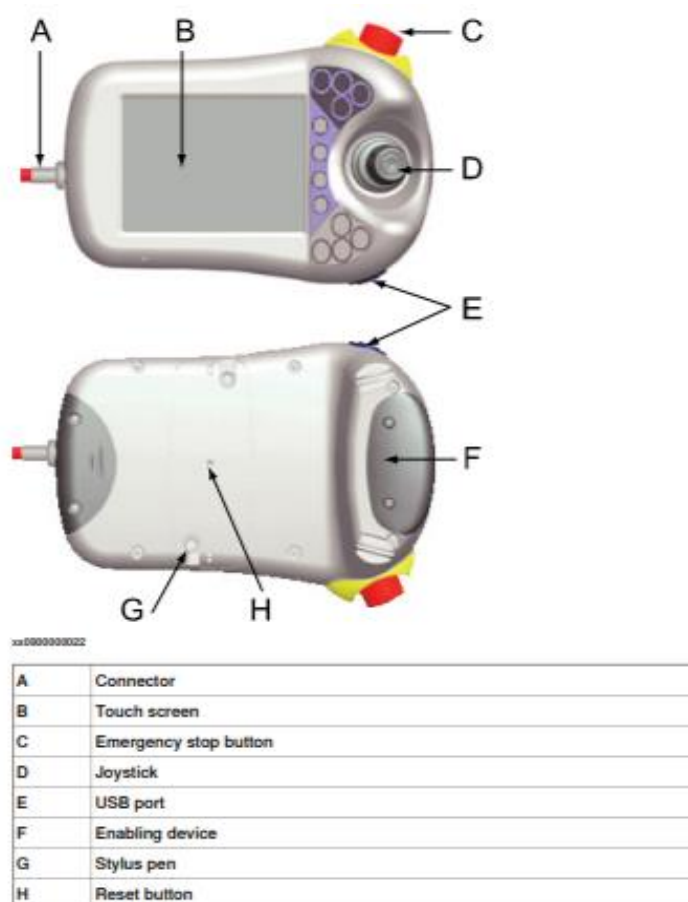


Figura 2.11. Imagen de la FlexPendant y la situación de sus principales elementos. (Fuente: Manual de instrucciones IRC 5)^[12]

2.4.4. Armario eléctrico

Junto a la cabina del controlador IRC 5 se encuentra el armario eléctrico. A continuación, se enumeran todos los elementos situados dentro del armario eléctrico:

- Fuente de alimentación de 24 V
- PLC Siemens S7-1515F
- Módulo de Entradas/Salidas, tanto para comunicación (Azul) como para las seguridades (Amarillo)
- Controlador CV-X150RP KEYENCE
- Líneas de distribución
- Terminales de conexión
- Magnetotérmicos
- Fusibles
- Filtros
- Termómetro
- Toma de corriente 230 V

La entrada de tensión del armario eléctrico debe ser una conexión de tres fases más neutro de 400 V y 50 Hz de frecuencia. El cableado usado para el armario eléctrico permite una corriente de 16 A y 4 KVA. Además, la celda dispone de aire comprimido a 6 Bar y un cabal de 300 l/min.



Figura 2.12. Fotografía del interior del armario eléctrico. (Fuente: Vlad Zaharia)

2.4.5. PLC Siemens SIMATIC S7-1515F

Dentro del armario eléctrico también se encuentra el PLC Siemens SIMATIC S7-1515F junto a sus módulos de seguridad. La serie de controladores SIMATIC S7-1500 es la última generación de controladores de TIA Portal y de automatización.^[20] Este modelo en concreto ha sido elegido para este proyecto con el fin de buscar la mayor eficiencia y de usabilidad en la máquina. Además, se decidió integrar la CPU de Seguridad SIMATIC S7-1515F, con el fin de gestionar la seguridad de la celda, ya que esta requiere gran memoria de datos y programa.

Para permitir la conexión con el controlador IRC 5 y los periféricos de la celda, vía PROFISAFE, se utiliza la interfaz integrada de PROFINET. Las CPU S7-1515F está certificada para funcionalidad de seguridad según norma EN61508, y para aplicaciones hasta SIL 3 (IEC62061) y PL (ISO 13849).



Figura 2.13. Imagen del PLC Siemens S7-1515F. (Fuente: Vlad Zaharia)

2.4.6. Cámara de visión artificial

La cámara está ubicada en una plataforma fija en la parte lateral de la celda para proporcionar una vista general de la pieza. La función principal de la cámara es confirmar que el clip se introdujo correctamente en la pieza mediante la detección de objetos. El modelo de cámara es KEYENCE CV-200M con resolución de 2 MP B / N y lente CA-LH8 con punto focal a 8 mm, controlada por el controlador CV-X150RP. También hay un panel de iluminación fijo en la parte superior de la celda que asegurar repetitividad en el brillo de las imágenes tomadas, independientemente de la luz ambiental presente.



Figura 2.14. Imagen de la cámara CV-200M con la óptica CA-LH8. (Fuente: Manual de usuario CV-200M)^[15]



Figura 2.15. Fotografía del soporte de la cámara CV-200M. (Fuente: Vlad Zaharia)

2.4.7. Utillaje (Nido)

Para llevar a cabo el proceso de clipaje con éxito, se necesita que la pieza en la que se vaya a introducir los clips esté bien sujeta y fija a la estructura mientras el robot realiza esta acción. Para ello son necesarios unos utillajes que sean robustos y que no se deformen con facilidad. Estos utillajes también deben asegurar que el posicionamiento de piezas es repetitivo para asegurar que los orificios donde se introducen los clips están siempre en la misma posición, por lo tanto, habrá la mínima desviación posible.

Como en esta instalación se pueden ensamblar dos modelos distintos de piezas, ambas se ensamblan en el mismo nido o utillaje, ubicado en el lado de la celda. Este utillaje fue diseñado presentando una inclinación prácticamente vertical, mejorando así la maniobrabilidad del robot. Por este motivo el utillaje está equipado con las abrazaderas correspondientes para fijar la pieza con suficiente precisión. También hay sensores capacitivos en el nido para detectar la presencia de la pieza.

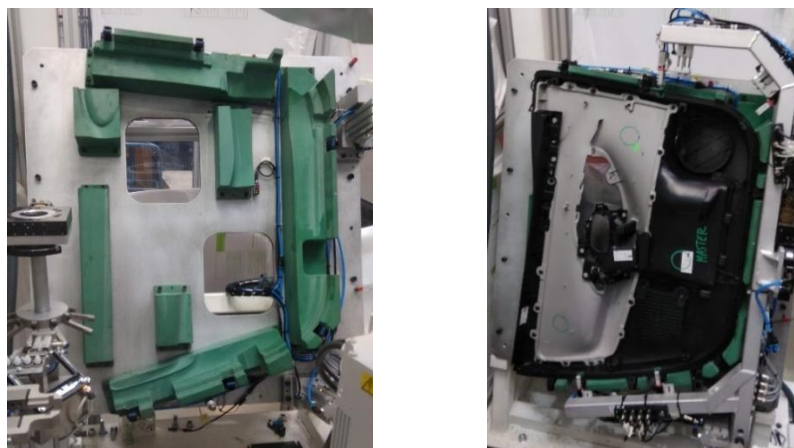


Figura 2.16. y Figura 2.17. Fotografía del utillaje de la estación de clipaje vacío (Izquierda) y junto con la pinza de manipulación y la pieza (Derecha). (Fuente: Vlad Zaharia)

También hay un utillaje dentro de la máquina de soldadura donde la pieza permanece hasta que el personal la retira.



Figura 2.18. Fotografía del utillaje de la máquina de soldadura. (Fuente: Vlad Zaharia)

2.4.8. Herramientas

La celda dispone de dos herramientas principales para el acoplo con el robot. El sistema que permite el intercambio de pinza al robot es un sistema comercial de la marca SCHUNK. La mitad de este sistema es acolado de forma permanente al sexto eje del robot, y la otra mitad se acopla a la herramienta. Además, las conexiones eléctricas de este sistema permiten al controlador del robot conocer en todo momento cual es la herramienta que sostiene el robot, o si en algún momento se pierde el contacto entre ambos.



Figura 2.19. Fotografía del sistema de cambio de herramienta SCHUNK. (Fuente: Vlad Zaharia)

La primera herramienta (tA en el programa) es un conjunto de pinzas que se usa para manipular la pieza. Dispone de dos pinzas estáticas y 2 pinzas dinámicas que se desplazan con la ayuda de unos cilindros neumáticos. Su función es poder asegurar la firmeza del agarre de la pieza en el proceso de manipulación de esta. Las piezas mecánicas fueron diseñadas con previsión permitiendo la posibilidad de ajustar la posición de las pinzas tridimensionalmente con la finalidad de encontrar la posición en la que se ejerza menor presión sobre la pieza plástica evitando así cualquier deformación o marca física.



Figura 2.20. Fotografía de las pinzas de la herramienta de manipulación. (Fuente: Vlad Zaharia)

La herramienta de clipaje (tB en el programa) está formada por 9 dedos de acero adaptados para el agarre de clips plásticos y tiene la función de introducir los clips en la pieza.

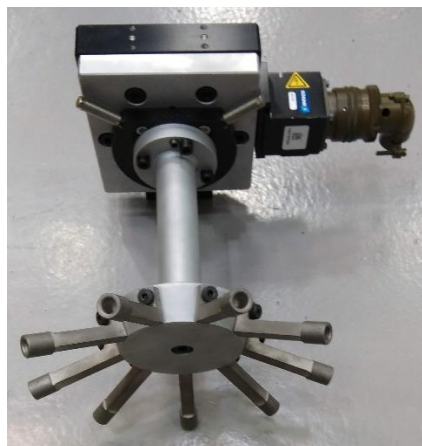
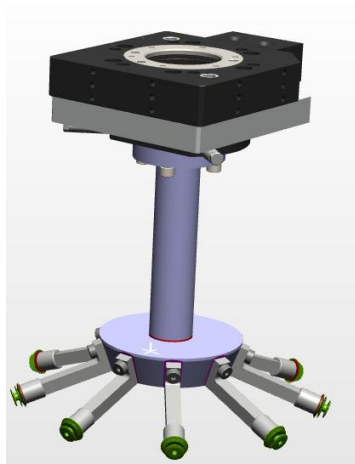


Figura 2.21. y Figura 2.22. Captura del modelo 3D (Izquierda) y fotografía (Derecha) de la herramienta de clipaje
(Fuente: Vlad Zaharia)

La cantidad de clips que se necesitan para ensamblar el Modelo 1 son 8 y para el Modelo 2 son 9 clips.

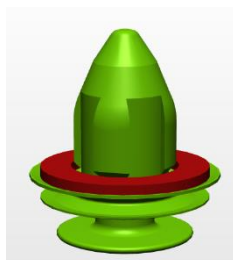


Figura 2.23 y Figura 2.24. Captura del modelo 3D del clip (Izquierda). Fotografía de un clip (Derecha). (Fuente: Vlad Zaharia)

2.4.9. Alimentadores de clips

La celda incluye un alimentador de clips plásticos ubicado en la parte posterior de la celda. El alimentador tiene una cadencia de 20 clips / min, y tiene una autonomía de 2 horas. Hay una salida lineal única y horizontal.

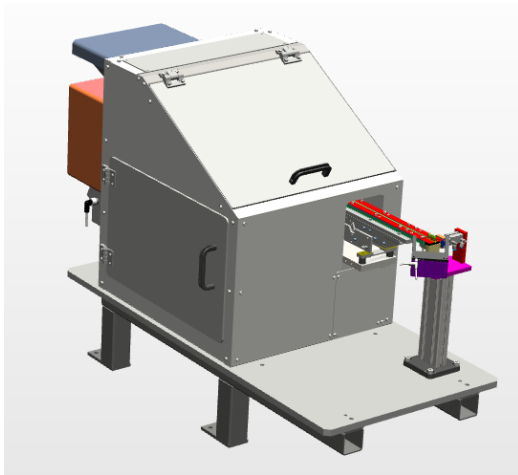


Figura 2.25. y Figura 2.26. Modelo 3D del alimentador de clips (Izquierda) y fotografía del interior del tambor de vibración. (Fuente: Vlad Zaharia)

El mecanismo de alimentación de los clips dispone de un dispensador lineal, por lo tanto, los clips se cogen uno a uno.



Figura 2.27. Fotografía del dispensador lineal de alimentación de clips. (Fuente: Vlad Zaharia)

2.4.10. Sistema de limpieza

Cuando alguno de los clips no se coloca en la pieza, el sistema detecta que el clip no se colocó y genera automáticamente un ciclo de degradado que consiste en limpiar el clip del dedo con una pinza. Este ciclo se ejecuta antes de que el robot coja clips en el siguiente ciclo de producción. Esto asegura que cuando el robot vaya a cargar el siguiente ciclo no tenga el clip en la herramienta y no haya colisión con el dispensador de clips.



Figura 2.28. Fotografía de la pinza de limpieza de clips. (Fuente: Vlad Zaharia)

2.4.11. Sistema de autoeyección

El sistema de autoeyección tiene el propósito de expulsar la pieza una vez que el ciclo de clipaje haya terminado. Este proceso mejora la comodidad de cambiar la pieza, lo que proporciona resultados más rápidos. El sistema de expulsión automática eleva la parte ya clipada y permite al personal colocar una parte nueva en el utillaje tras extraer la pieza terminada.

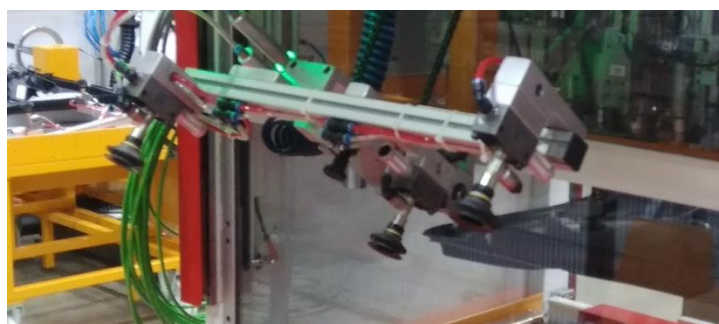


Figura 2.29. Fotografía del sistema de autoeyección. (Fuente: Vlad Zaharia)

2.4.12. Pantalla HMI

En la parte lateral de la celda se encuentra la pantalla HMI, que permite al personal supervisar y controlar la producción en todo momento además de determinar numerosas características de la producción. El modelo de la pantalla táctil es Siemens Confort Panel KTP900 TFT en color, y tiene un sistema mecánico de subida y bajada que permite al operario mover la pantalla tridimensionalmente a una posición cómoda de uso.



Figura 2.30. Fotografía del sistema Up&Down y la pantalla táctil igual a la usada en esta estación. (Fuente: Vlad Zaharia)

2.5. Elementos de seguridad

Para evitar que una persona acceda al interior de la celda y esta siga funcionando se deben añadir elementos externos de seguridad que o bien impidan la entrada de manera física a la persona o bien detenga el robot. Para ello se consideran principalmente las normas EN ISO 10218-1^[2] y 10218-2^[3], que son el estándar internacional de requisitos de seguridad y evaluación de riesgos de robots industriales. En estas normas aparecen indicaciones de carácter general sobre cómo debe ser un sistema robotizado teniendo en cuenta aspectos mecánicos, eléctricos, ergonómicos y de control. La norma que se sigue principalmente por el resto de los elementos es la EN ISO 12100^[4] que es la norma de seguridad de las máquinas, donde se encuentran indicaciones similares, pero de carácter más general. En la normativa europea UNE EN 775 se describe una estrategia de trabajo para seleccionar los elementos de seguridad. Los requisitos descritos en las normas tienen la función de reducir tanto el riesgo a personas, como aumentar la durabilidad de los elementos de la instalación

En el Anexo A1 se aprecian los pasos y las normas que deben seguir los usuarios de la celda en los distintos procesos que deban efectuar.

2.5.1. Seguridad PLC Siemens S7-1515F

En el caso de esta celda se ha preferido gestionar las seguridades de la celda desde el PLC, por lo tanto, todas las señales de seguridad de la celda, tanto provenientes del controlador IRC5, vía PROFINET, como de los elementos de seguridad de la celda, son procesadas por el PLC.

Así pues, existe una constante comunicación entre el IRC 5 y el PLC con el fin de supervisar las seguridades, tanto provenientes de la propia ejecución del programa como de la supervisión de movimiento del SafeMove. Aunque es el controlador que ejecuta de forma interna la supervisión del SafeMove, es el PLC que toma las medidas necesarias en caso de recibir alguna señal de paro de la celda.

Las paradas habituales que puede sufrir la instalación son:

- **Paro de Emergencia (ES)**
- **Paro General (GS)**
- **Paro Automático (AS)**
- **Paro de Programa (Stop)**

En el momento en que se produce alguno de estos paros, el robot detiene instantáneamente su movimiento y las válvulas de aire secundarias (que suministran aire a las herramientas del robot) se bloquean, para evitar la activación de los cilindros o los elementos neumáticos de la herramienta. Se

asegura que no se produzcan movimientos inesperados dentro a la celda debido al corte del flujo de aire. Estos movimientos podrían provocar la caída de algún elemento o de la pieza manipulada, pudiendo causar un coste económico añadido.

El SafeMove envía dos salidas digitales, una de **Paro General** y otra de **Paro Automático**, al PLC a través del PROFISAFE. Tanto en caso de **Paro General**, como si se produce algún **Paro de Emergencia** causado por las setas de emergencia de la celda, o cualquier otra razón establecida por el PLC, se pierde el puntero del programa del controlador y es necesario volver a las condiciones iniciales para poder continuar con la producción. Si se solicita el acceso a la celda en una de las puertas o se presiona el **Paro de Programa** el PLC detiene el ciclo de producción, pero el controlador mantiene la posición de puntero para permitir la recuperación del ciclo una vez rearmada la celda.

2.5.2. Seguridad del robot IRB 2600 y el controlador IRC 5

El propio robot cuenta con medidas de seguridad incorporadas de serie. Estas medidas se centran en la protección del usuario, aunque algunas de ellas sirven para evitar daños en las unidades mecánicas. Las medidas que el robot tiene integradas de serie son:

- **Pulsadores (o setas) de paro de emergencia:** El controlador IRC 5 cuenta con dos pulsadores de emergencia, uno en el armario del controlador y uno en la unidad de programación, siendo así muy accesible al programador del robot. El controlador IRC 5 también permite conectar más pulsadores de paros de emergencia, conectándolos en serie en la tarjeta de seguridades del controlador. En el caso de esta celda se ha preferido gestionar las seguridades generalmente desde el PLC, por lo tanto, todas las señales de seguridad de la celda son procesadas por el PLC. Cuando el controlador del robot recibe una señal de paro de emergencia procedente del PLC o de uno de sus dos pulsadores, los motores dejan de recibir tensión y en consecuencia se activan los frenos mecánicos de todos los ejes del robot, incluyendo los ejes externos. Si se desea rearmar la celda se debe restablecer el canal de seguridad y activar los motores para recibir tensión y poder desbloquear los frenos. Una parada de emergencia obliga además a llevar el robot a las condiciones iniciales, impidiendo así la ejecución del programa en el punto en el que había parado.
- **Frenos de los motores:** El sistema de freno del robot está formado por un electroimán, que mantiene los frenos inactivos mientras recibe tensión y que permite actuar a los frenos cuando deja de recibir tensión. Este factor permite detener el robot cuando hay algún fallo, como por ejemplo una pérdida de tensión.
- **Pulsador de habilitación (o palanca “hombre muerto”):** La consola de programación (FlexPendant) tiene en uno de sus laterales un pulsador de hombre muerto de tres posiciones. En una de estas posiciones (la intermedia) se activan los motores del robot, y en las otras dos

posiciones estos se bloquean. En caso de que se exceda la fuerza de pulsación también se desactivaran los motores ya que podría tratarse de una pulsación involuntaria. Siempre que se desee ejecutar movimiento con el robot en el modo de funcionamiento manual, se debe mantener accionado el pulsador en su posición intermedia para tener los motores activados.

- **Supervisión del movimiento:** Es una medida de seguridad que incluyen todos los robots y que le obligan a detenerse cuando hay un aumento inesperado del par en uno o más motores. Este aumento suele significar que ha habido alguna colisión entre la unidad mecánica y un elemento externo que podría ser tanto a causa de un choque contra un obstáculo como que una persona haya quedado atrapada entre el robot y una parte fija. Cuando salta la supervisión cae la tensión de los motores y se activan los frenos. A diferencia de la parada de emergencia, en este caso sí se podría seguir el programa en el punto donde se ha producido la supervisión una vez eliminada la causa de la colisión.
- **Limitación de la velocidad máxima:** En modo de funcionamiento manual la velocidad lineal máxima del TCP se ve limitada 250 mm/s con el propósito de reducir la energía de un impacto indeseado.

Como se verá más adelante, aparte de las medidas de seguridad que vienen de serie con el controlador IRC 5, también es posible integrar nuevas medidas de seguridad, entre otras el sistema SafeMove.

2.5.3. Vallado

La celda está rodeada de vallas de policarbonato, que impiden que el personal acceda al interior de la celda sin solicitar permiso a través del botón de las puertas. Estas barreras deben dejar suficiente espacio de dentro de la celda para que el robot pueda efectuar de forma “cómoda” sus movimientos y además permitir el acceso a todas las estaciones de la celda a los técnicos de mantenimiento.

El lado frontal de la máquina de soldadura está abierto al acceso del operario, para permitirle interactuar con la pieza, retirándola y reemplazándola por una nueva.

2.5.4. Puertas y cerrojos

Durante las tareas de programación, mantenimiento, calibración, u otra acción similar que precise de la presencia de una persona dentro de la celda, es necesario instalar un método de entrada y salida de la celda, siempre garantizando la seguridad integral. La opción seleccionada en este proyecto es la incorporación de dos puertas corredera de acceso lateral, que además permiten introducir el carro de mantenimiento entre el robot y la máquina de soldadura.

Ambas puertas incorporan un cerrojo, con un sensor de cierre y un interruptor de seguridad con enclavamiento electromagnético. Este dispositivo tiene un electroimán que bloquea el mango cuando

está en posición de seguridad y sólo permite desbloquearse si se pulsa un botón que baja las seguridades y para el robot. Como existe la posibilidad de que alguien quede atrapado dentro de la celda cuando se cierran las puertas, se han añadido en el interior palancas antipánico (una por cada puerta) que al activarse permiten abrir la puerta, aunque esté la posición de seguridad activada y hacen saltar la parada de emergencia.

Los cerrojos cuentan con dos botones y un pulsador de paro de emergencia. El botón amarillo sirve para solicitar el acceso a la celda cuando el electroimán está activo. Un botón blanco en cambio sirve para rearmar la celda.



Figura 2.31. Imagen del cerrojo instalado en las puertas de acceso. (Fuente: Anexo A1 “User Manual”)

2.6. Ciclo de funcionamiento de la máquina

Para entender mejor el ciclo de trabajo de la máquina expuesta en este documento a continuación se describe de forma resumida el proceso de trabajo habitual de la máquina.

En primer lugar, el robot accede al interior de la máquina de soldadura de plásticos por ultrasonidos. Una vez dentro, si procede, depositará la puerta del ciclo anterior en la parte inferior de la máquina de soldadura. A continuación, recogerá la pieza plástica que se acaba de soldar de la parte superior de la máquina de soldadura. Seguidamente situará la pieza junto con la herramienta de manipulación en el utillaje de clipaje. En el momento de soltar la pieza se efectuará el disparo de la cámara para comprobar si hay algún clip ya colocado y así no volver a ponerlo durante el clipaje. Después de soltar la herramienta de manipulación el robot va a buscar la herramienta de clipaje. Si es necesario, irá a limpiar los clips que puedan estar presentes en la herramienta de clipaje para después alimentarla de clips nuevos. Tras situar todos los clips en la pieza, el robot suelta la herramienta de clipaje y vuelve a agarrar la herramienta de manipulación junto con la puerta acabada de clipar. En este instante se vuelve a realizar una captura de imagen, esta vez para comprobar si los clips se han colocado correctamente. Al finalizar, volverá a acceder al interior de la máquina de soldadura para soltar la pieza acabada de ensamblar y si procede, coger una pieza nueva.

A continuación, se muestra una tabla [Tabla 2.3] con el ciclo detallado que ejecuta la máquina teniendo en cuenta que empieza en condiciones iniciales de la máquina y finaliza con el último ciclo de producción. Desde el paso 1 hasta el paso 17 se considera el primer ciclo de la celda robotizada. Una vez haya realizado hasta el paso 31 la máquina volvería al paso 18 para seguir con el ciclo habitual, o continuaría con el paso 32 en caso de que sea el último ciclo de producción. En la tabla también se puede apreciar de forma aproximada el tiempo que tardaría en realizar cada paso del ciclo, siendo la suma del ciclo habitual (del paso 18 al 31) de 58 segundos, aunque el tiempo máximo establecido por el cliente es de 62 segundos.

| Paso | Tiempo (s) | Acción |
|------|------------|--|
| 1 | 0 | Robot en espera en la posición de HOME sin ninguna herramienta. |
| 2 | 1,5 | Coge la herramienta de manipulación, que al ser el primer ciclo no debería tener pieza. |
| 3 | 0,5 | Abre las pinzas de la herramienta de manipulación para evitar colisionar con la pieza que va a recoger. |
| 4 | 6,5 | Accede al interior de la máquina de soldadura y va a la posición superior para recoger la pieza nueva. |
| 5 | 0,5 | Cierra las pinzas de la herramienta de manipulación la sostener firmemente la pieza. |
| 6 | 6,5 | El robot se sitúa en al útil de clipaje. |
| 7 | 1,5 | Suelta la herramienta de manipulación junto con la pieza. |
| 8 | 0,1 | La cámara captura la fotografía para saber el estado previo de la pieza. |
| 9 | 4 | El robot coge la herramienta de clipaje. |
| 10 | 9 | Realiza la limpieza de clips (si es necesario). |
| 11 | 10 | Alimenta la herramienta de clipaje con clips |
| 12 | 13 | Ensambla los clips en la pieza situada en el utillaje. |
| 13 | 3 | Suelta la herramienta de clipaje en su almacén. |
| 14 | 1 | El robot se sitúa en la posición previa al agarre de la herramienta de manipulación y se realiza la fotografía de comprobación posterior al clipaje. |
| 15 | 1,5 | Agarra la herramienta de manipulación junto con la pieza. |
| 16 | 6,5 | Accede al interior de la máquina de soldadura, posicionándose en el útil inferior. |
| 17 | 0,5 | Abre las pinzas de la herramienta de manipulación, soltando así la pieza clipada. |
| 18 | 1 | Va la posición superior para recoger la pieza nueva. |
| 19 | 0,5 | Cierra las pinzas de la herramienta de manipulación la sostener firmemente la pieza. |
| 20 | 6,5 | El robot se sitúa en al útil de clipaje. |
| 21 | 1,5 | Suelta la herramienta de manipulación junto con la pieza. |
| 22 | 0,1 | La cámara captura la fotografía para saber el estado previo de la pieza. |
| 23 | 4 | El robot coge la herramienta de clipaje. |
| 24 | 9 | Realiza la limpieza de clips (si es necesario). |
| 25 | 10 | Alimenta la herramienta de clipaje con clips. |
| 26 | 13 | Ensambla los clips en la pieza situada en el utillaje. |
| 27 | 3 | Suelta la herramienta de clipaje en su almacén. |
| 28 | 1 | El robot se sitúa en la posición previa al agarre de la herramienta de manipulación y se realiza la fotografía de comprobación posterior al clipaje. |
| 29 | 1,5 | Agarra la herramienta de manipulación junto con la pieza. |
| 30 | 6,5 | Accede al interior de la máquina de soldadura, posicionándose en el útil inferior. |
| 31 | 0,5 | Abre las pinzas de la herramienta de manipulación, soltando así la pieza clipada. |
| 32 | 6,5 | Va al utillaje de clipaje. |
| 33 | 1,5 | Suelta la herramienta de manipulación. |
| 34 | 0,5 | Vuelve a la posición de HOME sin ninguna herramienta. |

Tabla 2.3. Pasos y acciones detalladas del ciclo de la celda robotizada (Fuente: Vlad Zaharia)

2.7. La simulación 3D

Como se ha mencionado anteriormente, con la ayuda del programa RobotStudio se ha creado una estación virtual que simula la solución real de la instalación. En este modelo virtual se pueden observar los diferentes sistemas que componen la celda y posteriormente se ha procedido a simular los movimientos necesarios para llevar a cabo el ciclo de producción. Tras montar la estación a partir de los modelos 3D de los componentes suministrados por los proveedores, se ha comprobado que la distribución es correcta y que el robot alcanza correctamente todas las posiciones necesarias para el ciclo de producción. Este proceso tiene mucha importancia en la industria actual, ya que en este mundo cada vez más globalizado, la industria se vuelve cada vez más competitiva y busca el mayor beneficio al menor coste. El proceso de simulación 3D mejora notablemente el rendimiento de los sistemas, gracias a las tareas de rediseño de los prototipos. Esto ocurre cuando la estación es simulada y permite observar que algunos elementos no son los adecuados para la solución diseñada. Además, este es un factor determinante en el coste económico del proyecto, ya que se evitan costes producidos a raíz de la fabricación o la compra de componentes que no encajan correctamente en la solución, además de la posibilidad de optimizar el ciclo de producción antes de implementar el programa en el robot.

A continuación, se detallan los pasos seguidos en el proceso de la creación de la simulación. Si se desea conocer la metodología de trabajo y como se realizan las distintas acciones para crear la estación 3D y posteriormente elaborar una simulación, en el Anexo C1 aparece una “Guía rápida de simulación”.

2.7.1. Creación de la estación virtual

El primer paso para la creación de la simulación es la creación de la estación y el posterior ensamblaje de los modelos 3D de los componentes de la celda. Este proceso se inicia después de determinar una solución por parte del equipo del proyecto, con la intención de poder representar la distribución planteada y así mostrar al cliente, junto con la oferta, como sería la celda. Los modelos 3D de cada componente son facilitados por los proveedores de estos, y posteriormente ensamblados por un experto del equipo técnico de ABB. Las posiciones de cada componente se determinan según los estudios y diseños previamente realizados, y a la hora de crear la estación virtual se mantienen tanto las posiciones como las orientaciones especificadas en los planos mecánicos. A pesar de esto, si es necesario desplazar algún componente debido a una colisión entre objetos o para mejorar la distribución, esta decisión es comunicada al equipo de diseño, siendo entonces una acción realimentada, ya que este es uno de los propósitos de la estación virtual.

Esta tarea la realiza un Ingeniero mecánico experto en creación de simulaciones 3D debido a que se precisa de gran velocidad para poder mostrarle al cliente la solución planteada lo antes posible. Además, este ha recibido soporte por parte del autor de esta memoria a la hora de colocar algunos

componentes (base del robot, alimentador, armarios, etc.). Este proceso es continuo durante todo el proyecto, ya que antes de realizar cualquier cambio en los elementos de la estación, previamente se implementa en la estación 3D para comprobar su viabilidad. Así pues, se puede apreciar una evolución en el conjunto 3D a lo largo de todo el proyecto.

De forma paralela a la simulación creada por el exporto en creación de simulaciones 3D, el autor de este trabajo ha desarrollado desde el inicio una segunda estación virtual y su posterior simulación, con el fin de mantener actualizada la estación virtual durante el avance del proyecto, ya que en algunas ocasiones estas modificaciones se dejan de aplicar a la estación virtual con el fin de ahorrar recursos humanos. Además, este ejercicio ha servido para la elaboración del presente documento y el aprendizaje autónomo de los procesos necesarios para su desarrollo.

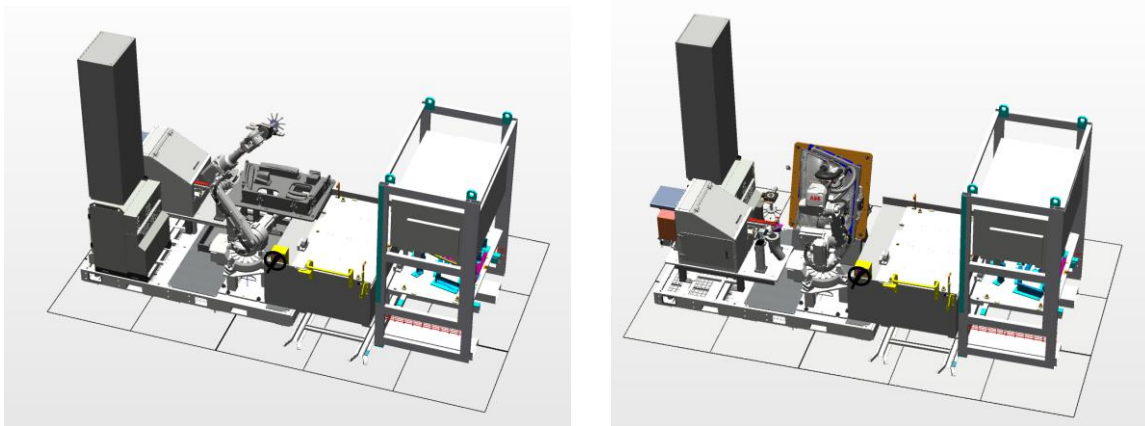


Figura 2.32. y Figura 2.33. Comparación entre la estación 3D inicial (Derecha) y la final (Izquierda). (Fuente: Vlad Zaharia)

2.7.2. Preparación de la simulación

En este paso del desarrollo, lo que se pretende es preparar virtualmente la estación para poder iniciar la simulación de los movimientos.

En primer lugar, se precisa crear las relaciones y conexiones entre los distintos objetos. Dos ejemplos claros son el hecho de acoplar el paquete energético (DressPack) al eje 3 del robot, o el acople del cambio de herramientas (SCHUNK) al eje 6 del robot, para poder después acoplar las herramientas de manipulación y clipaje a él.

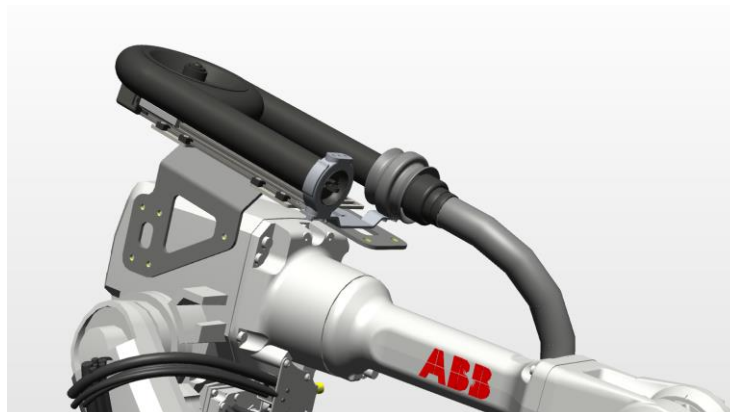


Figura 2.34. Captura del modelo 3D del paquete energético acoplado al eje 3 del IRB 2600. (Fuente: Vlad Zaharia)

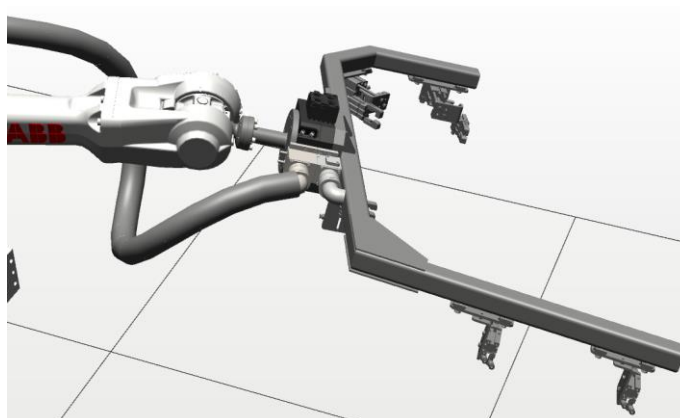


Figura 2.35. Captura del modelo 3D del cambio de herramienta y la herramienta de manipulación acoplados al eje 6 del IRB 2600. (Fuente: Vlad Zaharia)

Una vez determinadas las relaciones entre los elementos tridimensionales se procede a crear las herramientas y los objetos de trabajo, para poder así referenciar de forma correcta los puntos y los movimientos a los sistemas de coordenadas deseados.

Otro paso de la preparación de la simulación es la creación de las posiciones (puntos, posiciones de ejes) para su uso en las trayectorias de la simulación. Este proceso no es definitivo ya que durante la creación de los movimientos los puntos suelen ser modificados o sustituidos por otros.

2.7.3. Creación de las trayectorias

La herramienta de simulación de RobotStudio permite la creación de las trayectorias de una forma sencilla e intuitiva. El método más ágil para crearlas consiste en determinar las trayectorias arrastrando los puntos por los que se desea que pasen en el orden del movimiento. Posteriormente, estos movimientos se transmiten al programa RAPID, mediante la sincronización de la estación y el programa del controlador virtual, para el ajuste de sus características (velocidades, herramientas, objetos de trabajo, reducciones de zona, etc.).

Es necesario comentar que las velocidades de los movimientos dependen del peso y el tamaño de la herramienta actual en uso, de la distancia de estos y de la precisión que se desee obtener. Si se trata de movimientos relativamente largos, en los que el robot no tiene ninguna herramienta, o la herramienta de clipaje que tiene un peso aproximado de 3 kg, estos movimientos se realizan a la máxima velocidad que ofrece el robot. Una vez el robot llegue al punto deseado y se quiera que realice un movimiento corto y preciso, maximizando la seguridad, se disminuye mucho la velocidad para que el robot acceda al punto fino sin llevar inercia. Hay que tener especial cuidado cuando se use la herramienta de manipulación junto con la puerta debido a su gran tamaño y elevado peso (8 kg sin carga, 11 kg junto con la puerta). Este factor puede afectar sobre todo cuando sea necesario realizar movimientos de reorientación ya que toda la inercia que produce este movimiento recae en la mecánica de los ejes más pequeños del robot (ejes 4, 5 y 6). Por esta razón, como se puede apreciar en la simulación, también se reduce mucho la velocidad cuando se realizan los giros de la herramienta de manipulación.

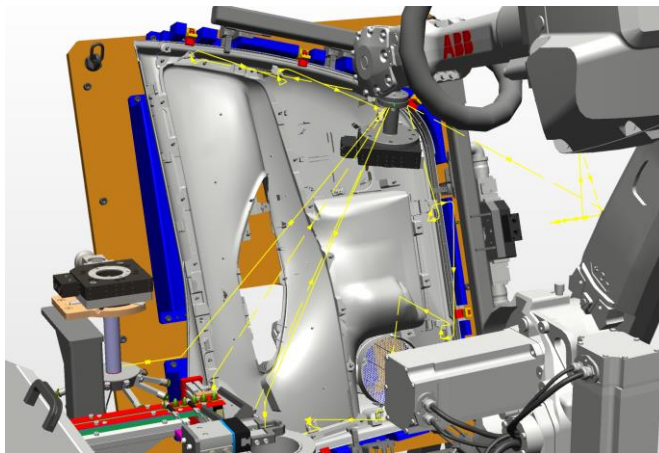


Figura 2.36. Captura de las trayectorias (amarillo) realizadas por el robot IRB 2600 en la estación virtual.

(Fuente: Vlad Zaharia)

2.8. Programación básica de la simulación

Una vez se tenga el módulo de programa que agrega todos los movimientos previamente creados, es el momento de estructurar el programa. El programa elaborado para la simulación es sencillo y breve ya que su función principal es que se produzcan los movimientos de forma correcta en la estación virtual. En este proceso se ha obviado toda la parte lógica del programa ya que esta retrasaría el tiempo de creación de la simulación.

En primer lugar, se deben separar los movimientos por rutinas, y crear una rutina principal que las ejecute por el orden deseado. También se crean las señales necesarias para las interacciones entre los objetos de la simulación (visibilidad de la pieza y los clips). Estas señales se pueden usar una vez implementado el programa al controlador real del robot para la comunicación con el PLC y los periféricos de la celda.

Por último, se crea otro módulo (CalibData) donde se colocan todos los datos relacionados con la definición de herramientas y objetos de trabajo, con el fin de separar estos datos del resto del programa.

Cabe mencionar que tras finalizar el programa usado en la simulación se adaptan las declaraciones para seguir la nomenclatura estándar en las celdas de estas características.

A continuación, se muestra algunos fragmentos del módulo principal (MainModule):

```
MODULE MainModule
  !Módulo principal

  !Declaración de los puntos necesarios

  !Puntos Auxiliares
  CONST robtarget pAux10:=([-50.074540723,639.417996251,880.479953182],[0.073331542,-
0.978072924,0.193195433,0.025910721],[1,-1,-1,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]);
  !Puntos de manipulación de pieza e interacción con utillajes
  CONST robtarget pHandling10:=([-1508.87837479,830.265776671,224.634744785],[0.405579673,-
0.579227996,-0.405579819,0.579227993],[0,0,1,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]);
  CONST robtarget pHandling20:=([-1573.430911011,757.521634512,421.956370147],[0.396064797,-
0.624224348,-0.487413533,0.464655451],[0,0,1,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]);
  CONST robtarget pHandling30:=([-1573.430000762,757.519995569,556.84004917],[0.39605595,-
0.624211741,-0.487398344,0.464695859],[0,0,1,0],[9E+09,9E+09,9E+09,9E+09,9E+09,9E+09]);
```

En este primer fragmento se aprecia la declaración de algunos de los puntos usados en el programa, siendo estos tipos de datos robtarget o jointtarget. Estos mismos puntos se pueden usar en el controlador real una vez se le cargue el programa, ya que al conservar la fidelidad de la parte mecánica

de ambas estaciones (virtual y física) las posiciones serán prácticamente las mismas. Aun así, será necesario volver a gravar las posiciones para ajustarse a la perfección a los elementos físicos de la celda.

```
PROC main()
```

```
    !Reset y Set de las distintas señales (En la simulación se usan para hacer ciertos elementos visibles o invisibles según convenga.
```

```
    !Así mismo, se usarán más adelante en la simulación para comunicar con el PLC
```

```
    Reset do_tA;
    Reset do_tB;
    Reset do_tB_Welding;
    Set do_tB_AutoEject;
    Set do_tB_Clip1;
    Set do_tB_Clip2;
    Set do_tB_Clip3;
    Set do_tB_Clip4;
    Set do_tB_Clip5;
    Set do_tB_Clip6;
    Set do_tB_Clip7;
    Set do_tB_Clip8;
    Reset do_tB_Inspection;
```

```
    !Distintos procesos/rutinas que realiza en el ciclo de simulación
```

```
    MoveAbsJ jAux10,v1000,z100,tool0\WObj:=wobj0;
    rHandling;
    rPick_tA;
    rCleaning;
    rPickClips;
    rClipping;
    rDrop_tA;
    MoveAbsJ jAux10,v1000,z100,tool0\WObj:=wobj0;
```

```
ENDPROC
```

En este segundo fragmento aparece el proceso principal del programa, y es donde se sitúa el puntero del programa al iniciar la ejecución. En este proceso primero se inicializan las señales usadas tanto en la simulación como posteriormente en el programa del robot. A continuación, se ejecutan las rutinas que debe realizar el robot siguiendo el ciclo de la máquina.

```

PROC rCleaning()
  MoveAbsJ jAux10,v1000,z100,tool0\WObj:=wobj0;
  Confl\On; !Se utiliza para que el robot pueda alcanzar las posiciones usando la configuración
  de ejes más adecuada.
  MoveL pCleaning_Aux,v1000,z100,tA1\WObj:=wobjCleaning;
  MoveL pCleaning,v300,fine,tA1\WObj:=wobjCleaning;
  WaitTime 0.25;
  Reset do_tA_Clip1;

  MoveL pCleaning_Aux,v1000,z100,tA1\WObj:=wobjCleaning;
  MoveL pCleaning_Aux,v1000,z100,tA2\WObj:=wobjCleaning;
  WaitTime 0.25;
  MoveL pCleaning,v300,fine,tA2\WObj:=wobjCleaning;
  WaitTime 0.25;
  Reset do_tA_Clip2;

  MoveL pCleaning_Aux,v1000,z100,tA2\WObj:=wobjCleaning;
  MoveL pCleaning_Aux,v1000,z100,tA3\WObj:=wobjCleaning;
  WaitTime 0.25;
  MoveL pCleaning,v300,fine,tA3\WObj:=wobjCleaning;
  WaitTime 0.25;
  Reset do_tA_Clip3;

  MoveL pCleaning_Aux,v1000,z100,tA3\WObj:=wobjCleaning;

```

Este tercer fragmento pertenece al proceso de limpieza de la pinza. Es interesante destacar que en todos los movimientos se usan los mismos dos puntos (pCleaning y pCleaning_Aux), aunque la diferencia se encuentra en la herramienta que se usa como TCP para ejecutar este movimiento (tA1, tA2, tA3, etc.). Estas herramientas pertenecen a los distintos dedos de la pinza de clipaje. Es solo un ejemplo de las numerosas técnicas que se usan para simplificar el programa.

A continuación, se muestra algunos fragmentos del módulo de datos (CalibData):

```

MODULE CalibData
  !Módulo donde están declarados los datos de calibración (Herramientas y objetos de trabajo)

  !Herramientas (Tools)

  TASK PERS tooldata tA0:=[TRUE,[-0.002,0.001,353.016],[1,0,0,0],[1,[0,0,1],[1,0,0,0],0,0,0]];
  TASK PERS tooldata tA1:=[TRUE,[104.398,0.00150436,371.087],[0.705251,-0.193139,-
  0.0511612,0.680221]],1,[0,0,1],[1,0,0,0],0,0,0]];

```

Este fragmento muestra un ejemplo de cómo se declaran las herramientas de trabajo del robot, siendo estos tipos de datos tooldata.

!Objetos de trabajo (Work Objects)

```
TASK PERS wobjdata wobjFeeder:=[FALSE,TRUE,"",[[-638.999,319,461],[1,0,0,0]],[[0,0,0],[1,0,0,0]]];  
TASK PERS wobjdata wobjCleaning:=[FALSE,TRUE,"",[[-  
524.869,301.69,350.944],[1,0,0,0]],[[0,0,0],[1,0,0,0]]];  
  
ENDMODULE
```

Este fragmento muestra un ejemplo de cómo se declaran los objetos de trabajo del robot, siendo estos tipos de datos wobjdata.

Al terminar la elaboración del programa, se sincroniza la estación virtual con el controlador virtual para poder posteriormente ejecutar el programa de la misma manera como se ejecutaría en la instalación real. Se ejecuta el programa paso por paso para revisar todos los movimientos y ajustar las posiciones o las características del movimiento, con el fin de conseguir una simulación fiel al ciclo de trabajo deseado.

2.8.1. Creación de ejecutables

Una vez finalizado el proceso de simulación, esta es exportada a un ejecutable (.exe) con el fin de poder reproducirla en cualquier equipo independientemente de si tiene o no instalado el programa RobotStudio.

Este ejecutable permite visualizar en detalle el modelo 3D de la estación virtual, dando gran libertad al usuario, y también permite reproducir la simulación en caso de que se deseen también exportar los movimientos. Estos ejecutables permiten analizar de forma exhaustiva los tiempos del ciclo de trabajo de la celda.

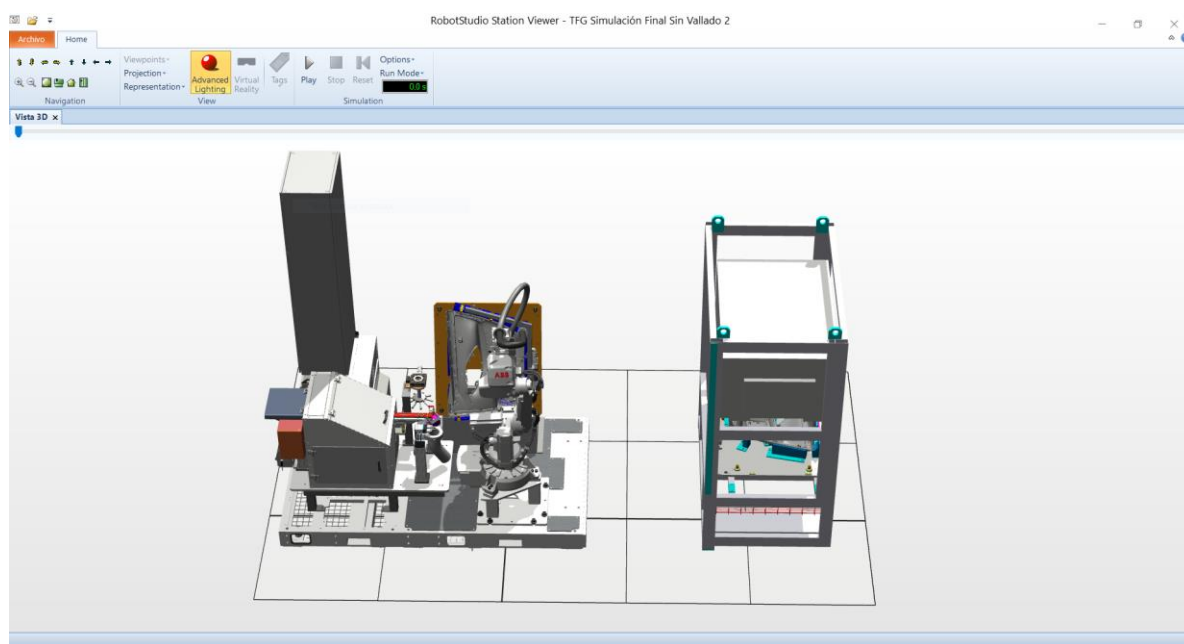


Figura 2.37. Interfaz del ejecutable de la simulación. (Fuente: Vlad Zaharia)

2.9. Estudio del tiempo de ciclo

Para realizar el estudio del tiempo de ciclo se crea una hoja de cálculo en la que se describen de forma muy detallada todos los pasos realizados por el robot y los segundos necesarios para ellos. Estos tiempos se obtienen tras analizar en profundidad el ejecutable de la simulación del proceso.

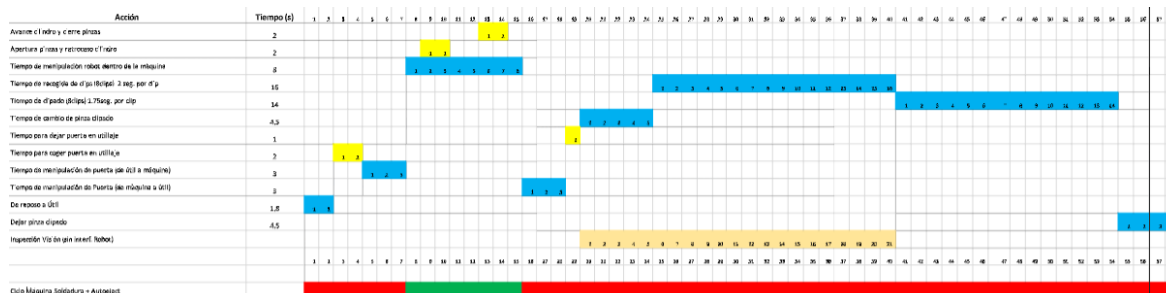


Figura 2.38. Captura del estudio del tiempo de ciclo obtenido de la simulación. (Fuente: Vlad Zaharia)

De esta forma, obtenemos un tiempo de ciclo total de 58 segundos, mientras el tiempo de ciclo requerido por el cliente es de 62 segundos. Para obtener un tiempo de ciclo de 58 segundos, por debajo del límite requerido por el cliente, se han optimizado las trayectorias en la simulación con el fin de realizar el menor número de movimientos y que estos sean lo más cortos posible. Dos ejemplos son los procesos de limpieza y alimentación de la herramienta de clipaje. En estos se puede observar que los movimientos que realiza el robot son muy cortos y se sigue un orden numérico en los dedos que son procesados. Hay que tener en cuenta que en esta simulación se realiza la limpieza de todos los dedos de la pinza de clipaje, mientras que en la realidad es algo muy poco común ya que lo habitual es que no se limpie ningún dedo, o uno o dos como mucho en caso de que no se haya llevado a cabo de forma correcta el proceso de clipaje. En lo que respecta al proceso de alimentación, aunque la herramienta de clipaje dispone de 9 dedos, únicamente se alimentan 8 de ellos ya que 8 es el número de clips que se ensamblan en la pieza.

Otro factor clave a tener en cuenta es que la interacción del operario no interfiere de ningún modo en el ciclo de la máquina. La única función del operario es retirar la pieza ya finalizada y sustituirla por una nueva. Esta acción la realiza mientras el robot esté clipando la pieza anteriormente introducida, de este modo cuando el robot vuelva al interior de la máquina de soldadura ya dispondrá de una pieza nueva a la que manipular.

Este estudio es entregado al equipo de programación del robot para que puedan seguir de forma pautada las operaciones descritas, facilitando así la tarea de comprensión del programa a realizar. El objetivo final es obtener unos tiempos de ciclo reales iguales o inferiores a los de la simulación, cumpliendo así los requisitos del cliente.

2.10. Desarrollo de la programación del controlador IRC 5

Una vez haya finalizado el proceso de simulación, y el programa creado para esta, ha sido adaptado, se procede a cargar este programa al controlador IRC 5 de la celda. En este momento se inicia la programación del robot. El proceso seguido es distinto a la programación de una simulación ya que las controladoras de los robots ya vienen con un sistema preinstalado, y lo que se hace es realizar cambios y modificar dentro de este sistema, en vez de cargar de forma directa el sistema de la estación virtual, ya que a este le podrían faltar algunas características que el controlador real sí tiene. Es un proceso mucho más complejo, y el programa es generosamente más elaborado que el de la simulación. Esto se debe a que el programa de la simulación es prácticamente una secuencia de movimientos que tienen la finalidad de recrear el ciclo de trabajo en la estación virtual, en cambio el programa realmente necesario para la instalación presenta una lógica compleja que dota la celda de autonomía. Esta autonomía, tanto tiene la función de ahorrar movimientos innecesarios como, por ejemplo, permitirle al robot volver a su posición de Home (posición inicial en la que tiene una configuración de ejes “cómoda”) en los casos en la que la celda ha sufrido un paro y se ha perdido el puntero del programa.

Se debe comentar que, al igual que en la programación del controlador de la visión por computadora, se debe prever el hecho de que se puede variar entre dos modelos distintos de pieza a la que clipar. Por lo tanto, aunque no se cree un segundo programa, ya que esto no es necesario, se deberá crear un segundo módulo de clipaje para el Modelo2. De esta forma, cuando el PLC le comunique por PROFINET al IRC 5 de que se produce un cambio de pieza, este ejecutará el mismo programa, con la única variación en el proceso de clipado de la pieza, ya que cambiarán las posiciones de algunos clips y se añadirá uno más, siendo en total 9.

El proceso de programación del controlador del robot ha sido confeccionado por el autor de este documento, en colaboración con un programador de una empresa subcontratada por ABB, que se especializa en los sistemas de gestión, automatización y robótica. Siguiendo así la recomendación de que durante la fase de programación de cualquier proyecto haya dos técnicos de programación, para que uno de ellos pueda pulsar la parada de emergencia en caso de que el segundo se encuentre en una situación de peligro mientras realiza su trabajo.

El papel que ha cumplido, el autor, en esta parte del proyecto, aparte de colaborar en la confección del programa, también ha sido la de transmitir al segundo programador el método de programación seguido por los programadores de ABB. Esta transmisión de conocimiento se ha conseguido gracias a la “Guía de programación” que pretende estandarizar el proceso de programación para las celdas del mismo cliente y de características similares.

Una recomendación a tener en cuenta a la hora de probar una sección del programa, que tiene una trayectoria nueva, es que esta se pruebe progresivamente, paso a paso, y a una velocidad reducida para evitar movimientos bruscos que puedan dañar la unidad mecánica o algún elemento de la celda. Después se puede ejecutar de manera continua y a una velocidad más elevada hasta llegar a la velocidad de trabajo.

En los siguientes apartados se especificarán los pasos seguidos para la programación del controlador IRC 5 y también se describirán algunas características y fragmentos del programa. No se pretende explicar al detalle las instrucciones, las rutinas o lógica del programa debido a que su extensión supera las 3 mil líneas de código, sino proporcionar los pasos seguidos para la elaboración de estos y su funcionamiento de forma general. Todas las partes del programa cuentan con comentarios que facilitan su entendimiento.

2.10.1. Guía de programación

Previamente al inicio de este proyecto, se reunió el equipo de programadores de ABB con el fin de iniciar una estandarización en la programación de las celdas de carácter similar, evitando así que cada programador siga sus propios procedimientos y nomenclatura en el programa. Se decidió crear un manual de programación que contenga el máximo posible de detalles relacionados con la programación, iniciando por la programación del robot, pasando por la programación de la visión artificial y terminando con la estandarización de las pantallas de los HMI. Esta tarea le fue encomendada al autor de la presente memoria, y junto a la ayuda de un programador experto, inició la elaboración de este manual.

Actualmente este manual se encuentra en una etapa inicial. A pesar de esto, dispone de la suficiente información para formar a un programador externo a ABB, en los pasos a seguir y que nomenclatura se debe respetar para facilitar el entendimiento del programa, tanto para el resto del equipo de programadores, como para los técnicos trabajadores de la empresa cliente.

Se puede decir entonces, que el propósito principal de esta guía es estandarizar la elaboración de los programas de las celdas del cliente de este proyecto, proponiendo algunas pautas a seguir en el proceso. En ella se describe la organización de las tareas y los módulos, cómo usar las rutinas, los argumentos que se necesitan para ellas, la nomenclatura que se debe usar y más detalles que pueden facilitar y acelerar el proceso de programación. Además, este documento podría utilizarse para comprender el funcionamiento de un programa ya existente que haya seguido las recomendaciones de la guía.

Con el objetivo de estandarizar la nomenclatura de los datos, se proponen algunas pautas a seguir a la hora de nombrarlos. Por lo general, el nombre de los datos comienza con la primera letra del tipo de

dato. Por ejemplo, los datos numéricos comienzan con la letra n (número), los datos de herramientas comienzan con t (tool), los datos de velocidad comienzan con v (velocidad), etc.

A continuación, se presenta una muestra de las recomendaciones que se dan a la hora de nombrar una posición. Se ha intentado dotar la explicación de mayor carácter visual con el fin de favorecer su comprensión y memorización.

Ejemplo 1:

pClipPick_F1_tA1_A10

p = Punto/Posición.

ClipPick = Acción que se realiza durante el movimiento.

F1 = El alimentador (Feeder)

tA1 = La herramienta (Tool)

A10 = El dedo.

Otro apartado clave de la guía de programación es el de los Primeros pasos, en el que se describen los pasos a seguir al iniciar la programación del controlador del robot. Este apartado sirve de referencia a los programadores principiantes asegurando que estos no se salten ningún paso necesario para desarrollar la programación de forma satisfactoria.

2.10.2. Primeros pasos

A continuación, se describen los primeros pasos necesarios para seguir cuando se inicia la programación de una celda:

En primer lugar, se debe guardar una copia de seguridad (BackUp) del estado del controlador al llegar de fábrica. Se recomienda realizar una copia de seguridad y almacenarla de forma externa, cada vez que se realice un cambio significativo, con la finalidad de poder revertir los cambios que deriven a algún error de programa o de ejecución. Para realizar las copias de seguridad se puede conectar el ordenador personal al puerto de servicio del IRC 5 mediante un cable Ethernet, y crear la copia de seguridad desde el programa RobotStudio, o simplemente introducir una memoria Flash USB a la consola de programación y realizar la copia de seguridad desde allí. Se recomienda mantener esta memoria junto al controlador durante toda la vida de la celda, para que en todo momento se pueda disponer de versiones anteriores a las que retroceder si así se precisa.

El segundo paso sería configurar la IP estática del controlador IRC 5 con el fin de no crear conflicto entre las IPs de los dispositivos conectados al PLC.

El tercer paso es cargar la lista de Entradas/Salidas (I/O) de algún proyecto anterior, o en el caso de que sea posible, una lista ya preparada para la propia instalación. En este caso se ha cargado la lista de

entradas y salidas de una celda de características similares, pero esta ha tenido que ser adaptada a las necesidades propias de la instalación.

Después de eso, se debe cargar el MOC de la simulación, de modo que se conozca la posición del robot en relación al mundo (o suelo en este caso). El MOC integra valores de las posiciones y orientaciones relativas de todas las unidades mecánicas de la celda.

Ahora es el momento de reiniciar el controlador del robot y comprobar que no aparezcan errores graves causados por los cambios previamente realizados.

El siguiente paso es cargar los módulos del programa que son necesarios para la celda. Dependiendo de la máquina, estos módulos serán diferentes. Estos módulos se pueden cargar de un proyecto de características similares o de una librería con módulos y rutinas estandarizados, creados para este tipo de celdas.

Después de eso, el programador debe analizar los módulos y las rutinas cargadas, borrar los que no sean necesarios y corregir los posibles errores que puedan aparecer.

A continuación, se carga el módulo de la simulación que contiene los movimientos. También se mueven los valores de las herramientas y los objetos de trabajo usados en la simulación al módulo CalibData del programa del controlador. Es el momento de cambiar el nombre de las herramientas, los objetos de trabajo y de los puntos.

Luego es necesario organizar los movimientos obtenidos de la simulación en los módulos a los que pertenecen (ChangeTool, CleanTool, PickClips, etc.).

El próximo paso es volver a definir las herramientas (TCP) y los objetos de trabajo con el uso de una herramienta especial que termina en punta. Este paso se realiza para darle mayor precisión al sistema en general. Tras la definición de las herramientas es necesario ejecutar la rutina de LoadID con el fin de obtener el peso y el centro de gravedad de cada herramienta.

Seguidamente se procede a situar el robot en todos los puntos guardados de la simulación, usando las herramientas y los objetos de trabajo recientemente definidos, y así volver a guardar las posiciones de forma más precisa.

El último paso es desarrollar y mejorar la lógica del programa para agregar más versatilidad a este.

2.10.3. Explicación de las tareas

Cada tarea dispone de un único puntero de programa, es decir, se ejecuta de manera secuencial y ninguna instrucción se puede ejecutar hasta que se haya terminado la ejecución de la instrucción anterior. Si se desea que se ejecuten varios códigos al mismo tiempo se necesitan más de un puntero de programa, y para hacer esto, se requiere más de una tarea. Por este motivo se incorpora la opción de multitarea en el controlador del robot, permitiendo la ejecución paralela de un máximo de 20 tareas diferentes.

Cada tarea dispone de sus propios módulos, y a su vez estos de rutinas. Esto significa que es posible tener módulos, rutinas o datos que se nombren de la misma manera en dos tareas distintas, ya que el puntero de cada tarea únicamente puede recorrer los módulos propios de la tarea.

Solo puede haber una tarea de movimiento para cada unidad mecánica gobernada por el controlador para evitar contradicciones entre las instrucciones de las diferentes tareas.

Una tarea puede ser de tipo Normal, Estática o Semiestática. y esta característica, junto al Trust Level (nivel de confianza) se definen en la configuración del controlador. Los valores posibles para el Trust Level son NoSafety, SysFail, SysHalt o SysStop. El nivel de confianza maneja el comportamiento del sistema cuando una tarea semiestática o estática se detiene o no se puede ejecutar.^[10]

Las tareas normales son las que se pueden activar y desactivar por el programador en cualquier momento desde la unidad de programación o el RobotStudio. Su ejecución se inicia en el momento en que se pulse el botón del Play, o el PLC envíe una señal de inicio de ejecución del programa. Las tareas en segundo plano, estáticas y semiestáticas, siempre se activarán automáticamente al encender el controlador. Además, estas solo se pueden detener si tienen el TrustLevel como NoSafety.

T_ROB1, donde 1 es el número del robot, incluye todas las rutinas e instrucciones relacionadas con el movimiento del robot y otras rutinas asociadas al mecanismo del robot. El tipo de ejecución de esta tarea es Normal y permite movimiento de robot.

T_EXTRA incluye las rutinas e instrucciones que no están incluidas en las otras tareas, y es necesario ejecutarlas en paralelo. En este proyecto se utiliza para un proceso que envía de forma continua al PLC un LifeBit, que indica que el controlador IRC 5 se encuentra encendido y en funcionamiento. El tipo de ejecución de esta tarea es semiestático y no permite movimiento de ninguna unidad mecánica.

2.10.4. Explicación de los módulos

Los módulos son fragmentos del programa en el que se sitúan los datos, los procesos y las funciones. Se suele organizar el programa en módulos con el fin de que cada módulo realice una función general distinta. Los módulos se pueden guardar, cargar y copiar como archivos independientes.

Los módulos se dividen en módulos de programa y módulos de sistema. Los módulos de sistema son los que se mantienen en la memoria del controlador, aunque se cambie de programa. En cambio, los módulos de programa son los que se sustituyen cuando se realiza un cambio de programa. En los módulos de sistema es donde se colocan las rutinas que se pueden usar independientemente del programa ejecutado, y también las declaraciones de la tool0 y el workobject0, siendo estos declarados de serie.

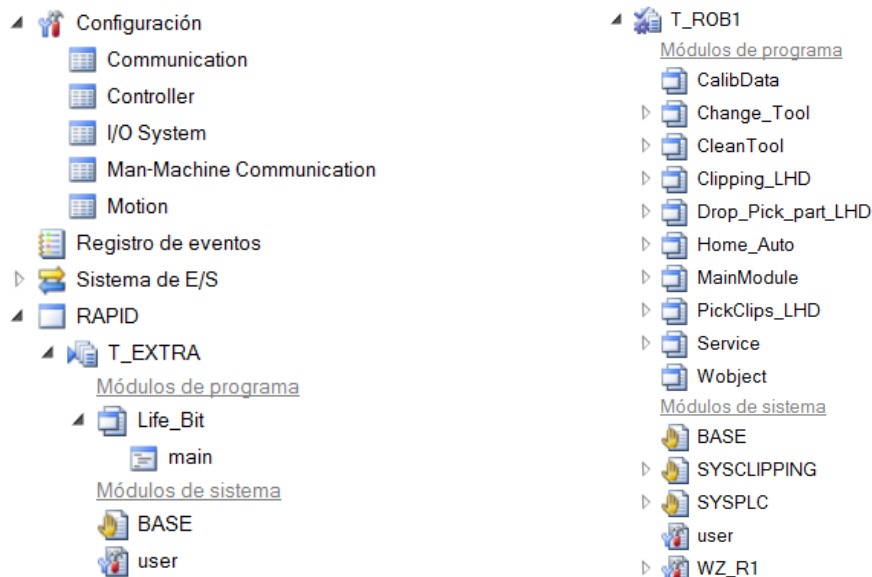


Figura 2.39. Estructura del programa RAPID. (Fuente: Vlad Zaharia)

Usualmente, se crea un módulo **CalibData** en el que se coloca la declaración de todas las herramientas y objetos de trabajo. En este proyecto se ha decidido colocar en él también la declaración de las posiciones, ya que su número es relativamente reducido y así se consigue mayor organización de los datos. Aunque en este proyecto solo se explica el desarrollo de una celda, como se ha comentado anteriormente, el proyecto realmente consta de la creación y programación de cuatro celdas idénticas, en las que lo único que cambia son el número y la posición de los puntos de clipaje. Por lo tanto, disponer de un módulo en el que se colocan todas las declaraciones, añade mucha versatilidad al proceso, porque es el único módulo que se debe modificar al pasar el programa de una celda a otra.

En el **MainModule** es el módulo principal de la tarea y es donde se ubica la rutina principal del programa (main), y es donde se sitúa el programa al iniciar la ejecución o al realizar un PP a Main. Desde este módulo se llaman las rutinas ubicadas en el resto de los módulos de la misma tarea. También es donde se sitúan todas las rutinas de inicialización del programa.

El módulo **Change_Tool** contiene las rutinas e instrucciones que ejecutan el proceso de cambio de herramienta. Este proceso consiste básicamente en dejar la herramienta actual en su almacén correspondiente y agarrar la otra herramienta o volver a HOME sin herramienta.

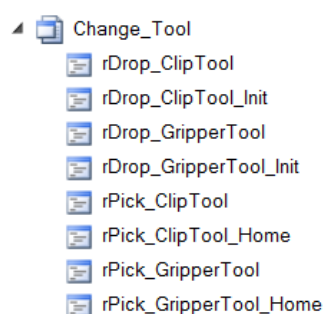


Figura 2.40. Contenido del módulo Change_Tool. (Fuente: Vlad Zaharia)

CleanTool es el módulo que contiene las rutinas e instrucciones que ejecutan el proceso de limpieza de la herramienta de clipaje. Este proceso consiste básicamente en quitar los clips del extremo de los dedos que se desee limpiar. El clip se introduce entre los dedos de la pinza de limpieza, de modo que cuando se cierra la pinza, y el robot se retira, el clip se separa del dedo.

Clipping, es el módulo que contiene las rutinas e instrucciones que ejecutan el proceso de clipaje de la pieza. Este proceso consiste básicamente en ensamblar los clips dentro de la pieza. Este módulo puede tener dos variantes, una para las piezas del Modelo1 y otra para el Modelo2.

Drop_Pick_Part, es el módulo que contiene los movimientos necesarios para el proceso en que el robot accede al interior de la máquina de soldadura y suelta la pieza ensamblada y recoge la pieza nueva.

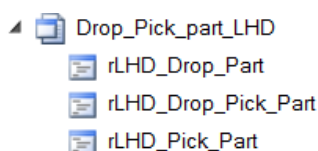


Figura 2.41. Contenido del módulo Drop_Pick_Part. (Fuente: Vlad Zaharia)

En el módulo **Home_Auto** se colocan todas las rutinas e instrucciones necesarias para que el robot sea capaz de volver a su posición de Home independientemente de la posición en la que se encuentre. Este

módulo es muy útil cuando se pierde el puntero de programa y se desea retomar la producción sin necesidad de situar el robot en la posición de Home de forma manual. En caso de que el robot este en una posición muy complicada para volver, se ha programado la emisión de un mensaje de error que solicita que el robot sea devuelto a su posición de forma manual.

El módulo **PickClips**, al contrario que el módulo de limpieza contiene las rutinas e instrucciones que ejecutan el proceso de alimentación de clips. Este proceso consiste básicamente en recoger los clips del alimentador con el extremo de los dedos de la pinza de clipaje. Esta acción dependería del alimentador en el que se desea recoger los clips, pero en el actual proyecto únicamente se utiliza un tipo de clips, por lo tanto, solo hay un alimentador.

En el módulo **Service** se encuentran todas aquellas rutinas que se deben ejecutar en caso de desear realizar alguno proceso de mantenimiento al robot.

El módulo de sistema **SYSCLIPPING** se encuentran las rutinas estandarizadas de ABB. Estas rutinas se utilizan de la misma manera que las instrucciones propias del lenguaje RAPID, como podría ser el ejemplo de un movimiento MoveL. Para ejecutar estas rutinas es necesaria la introducción de argumentos, como por ejemplo en la rutina CleanToolGrip, sería necesario indicar que herramienta y cuáles de sus dedos se desea limpiar.

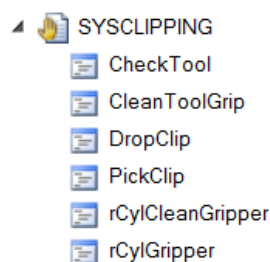


Figura 2.42. Contenido del módulo SYSCLIPPING. (Fuente: Vlad Zaharia)

SYSPLC contiene algunas instrucciones necesarias para la comunicación con el PLC.

Por último, encontramos el módulo **WZ_R1** que contiene los procesos de ejecución de las zonas mundo.

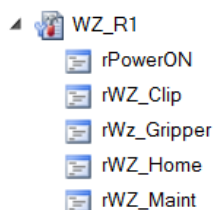


Figura 2.43. Contenido del módulo WZ_R1. (Fuente: Vlad Zaharia)

2.10.5. Explicación de las rutinas

Las rutinas se pueden dividir en tres categorías: procedimientos, funciones y rutinas TRAP.^[10]

- Los procedimientos se utilizan como subprogramas. Son el tipo más utilizado de rutina. En el caso específico del presente programa descrito, se usan procedimientos para ejecutar las instrucciones necesarias para cada acción.
- Las funciones devuelven un valor de un tipo concreto y se utilizan como argumento de una instrucción.
- Las rutinas TRAP proporcionan una forma de responder a las interrupciones de la ejecución secuencial. Las rutinas TRAP se asocian interrupciones determinadas con el fin de interrumpir la ejecución habitual del programa. El ejemplo más común es el uso de una entrada digital con el fin de interrumpir la ejecución del programa y proceder a ejecutar la rutina TRAP asociada.

Para ejecutar algunas rutinas se deben ingresar argumentos obligatorios u opcionales, con el fin de especificar los parámetros necesarios. Normalmente, los parámetros se usan sólo como datos de entrada y se tratan como variables de la propia rutina. Un parámetro puede ser opcional y puede ser omitido de la lista de argumentos de la llamada a una rutina. Los parámetros opcionales son precedidos por una barra invertida (\).

También es posible que dos o más parámetros opcionales sean excluyentes entre sí. En estos casos se indica mediante una barra vertical (|) entre los parámetros afectados.

Debido a la confidencialidad que presentan las rutinas usadas en este proyecto, no se revela el funcionamiento íntegro de ellas. Estas rutinas se suelen encriptar previamente a la entrega y puesta en marcha de la instalación.

A continuación, se muestran algunos fragmentos del proceso CleanToolGrip, que tiene la función de limpiar los dedos de la herramienta de clipaje que se le indique.

```

PROC CleanToolGrip(
  \switch L,
  robtarget ToPoint,
  speeddata vClean,
  num FrontSet,
  \num Backset,
  PERS tooldata tActive,
  \PERS wobjdata WObj
  \VAR signaldi diClip
  \VAR signaldi diClean)
  !-----
  !Switch L|J  -> primer movimiento en MoveL | MoveJ
  !ToPoint    -> target posición clip
  !vClean     -> velocidad de inserción
  !FrontSet   -> offset de entrada
  !Backset    -> offset de salida
  !tActive    -> tool en uso
  !WObj       -> se pueden programar trayectorias con o sin wobj
  !diClip     -> entrada para habilitar/deshabilitar solo la inserción del clip
  !diClean    -> entrada para habilitar/deshabilitar solo la inserción del clip
  !-----
  VAR num OffsetZ;
  VAR string stClip;

```

Como podemos observar en este fragmento se aprecian los argumentos, tanto obligatorios como opcionales para la correcta ejecución de esta. Así mismo, a continuación, aparece una explicación clara del significado de cada argumento. Seguidamente se declaran las funciones internas del proceso.

La estructura principal de esta rutina es un cúmulo de condicionales IF. Estos es así ya que según los argumentos que se hayan introducido se deberán realizar unas acciones u otras.

```

IF NOT Present(L) THEN
  MoveJ RelTool(ToPoint,0,0,FrontSet),vmax,fine,tActive\WObj:=WObj;
ELSE
  MoveL RelTool(ToPoint,0,0,FrontSet),vmax,fine,tActive\WObj:=WObj;
ENDIF
! * Open CylClean *
rCylCleanGripper\Open\bWaitOpen;
! * Clip In Finger *
! * If you are in manual mode, you can repeat *
IF OpMode()==OP_MAN_PROG THEN
  UIMsgBox\Header:="You want to clean your finger?","Clean the finger "+stClip+"."
  \Buttons:=btnYesNo\Icon:=iconInfo\Result:=answer;
  IF answer=resNo THEN
    RETURN ;
  ENDIF
ENDIF
ENDIF

```

En este fragmento en primer lugar vemos una condición en la que si se ha indicado como argumento opcional que se desea realizar el primer movimiento de forma lineal, entonces se ejecutará la

instrucción MoveL, usando el resto de argumento necesarios, como el punto, el offset, la velocidad, la herramienta y el objeto de trabajo.

Antes de proceder a introducir el dedo de la herramienta de clipaje entra los dedos de la pinza de limpieza se ejecuta una rutina que precisamente envía una señal al PLC para que se abra la pinza de limpieza, y permanece a la espera de la confirmación.

Si se ejecuta esta rutina de forma manual, aparece un mensaje en la consola de programación preguntando si el operario desea limpiar el dedo específico. Si es así procede a realizar los movimientos necesarios para la limpieza.

Estos movimientos se realizan en condición a los offsets de limpieza existentes, ya que según el tipo de clip que se quiera limpiar el modo de acceso o retirada es distinto.

```

IF present(diClip) OR present(diClean) THEN
  IF diClip=0 AND diClean=0 THEN
    ! * Clean Position *
    MoveL ToPoint,vClean,fine,tActive\WObj:=WObj;
    ! * Close CylClean *
    rCylCleanGripper\Close\bWaitClose;
    WaitTime 0.15;
    IF Present(Backset) THEN
      MoveL RelTool(ToPoint,0,0,Backset),vmax,fine,tActive\WObj:=WObj;
    ELSE
      MoveL RelTool(ToPoint,0,0,FrontSet),vmax,fine,tActive\WObj:=WObj;
    ENDIF
    ! * Open CylClean *
    rCylCleanGripper\Open\bWaitOpen;
  ENDIF
  IF diClip=1 OR diClean=1 THEN
    IF Present(Backset) THEN
      MoveL RelTool(ToPoint,0,0,Backset),vmax,fine,tActive\WObj:=WObj;
    ELSE
      MoveL RelTool(ToPoint,0,0,FrontSet),vmax,fine,tActive\WObj:=WObj;
    ENDIF
    ! * Open CylClean *
    rCylCleanGripper\Open\bWaitOpen;
  ENDIF
ELSE
  IF Present(Backset) THEN
    MoveL RelTool(ToPoint,0,0,Backset),vmax,fine,tActive\WObj:=WObj;
  ELSE
    MoveL RelTool(ToPoint,0,0,FrontSet),vmax,fine,tActive\WObj:=WObj;
  ENDIF
  ! * Open CylClean *
  rCylCleanGripper\Open\bWaitOpen;
ENDIF
RETURN ;
ENDPROC

```

2.10.6. Zonas mundo (World Zones)

Limitar las zonas en el espacio en las que el robot puede entrar y conocer en la que zona se encuentra, independientemente de la posición del puntero, es indispensable para el correcto funcionamiento de la celda. Hay dos opciones que ofrecen estas características, la de zonas mundo y el SafeMove.

Las zonas mundo estacionarias (WZstationary) permiten limitar las zonas por código dentro del programa del robot y obtener información del TCP del robot en el momento de ejecutarlas. Una zona mundo es supervisada durante los movimientos del robot, tanto durante la ejecución del programa como al movimiento manual. Si el TCP del robot llega a la zona mundo establecida, se puede detener el movimiento, o modificar el valor de una señal de salida digital. El inconveniente es que las zonas mundo quedan definidas en un módulo de programa que podría borrarse o modificarse fácilmente.

En este proyecto se decidió usar ambos métodos, tanto las zonas mundo como el SafeMove para conocer la zona en la que se encuentra el robot. La diferencia es que en el SafeMove únicamente se declararon dos zonas de gran tamaño, y en cambio, se han creado zonas mundo para cada estación de la celda (zona de clipaje, zona de alimentación, zona de limpieza, zona de máquina de soldadura, etc.).

Para declarar las zonas mundo hacen falta tres datos. En primer lugar, un shapedata, que determina la geometría de la zona. La geometría más usada en este programa es el punto ya que se desea establecer valores a las salidas digitales en el momento en que el robot pasa por un punto físico. El segundo dato necesario es el wzstationary. En combinación con el shapedata definen la zona mundo estacionaria.

```
!*** Home pos. WZ ***
! Use tool0, WObj0.
TASK VAR shapedata SHOME;
TASK VAR wzstationary WHOME;
TASK PERS jointtarget jdelta_posHome:=[[2,2,2,2,2],[9E9,9E9,9E9,9E9,9E9]];
```

Por último, se deben declarar las posiciones en las que se desea situar la zona mundo. Estas posiciones van asociadas comúnmente a un jointtarget (posición de ejes del robot) aunque también se puede usar el robtarget (coordenada tridimensional de la posición del TCP).

A continuación, se crea un proceso que incorpore los tres datos anteriormente mencionados. En el ejemplo, se establece que cuando el robot se encuentre dentro de la zona mundo declarada, este establecerá la salida digital a 1.

```
PROC rWZ_Home()
    WZHomeJointDef\Inside,SHOME,jHome,jdelta_posHome;
    WZDOSet\Stat,WHOME\Inside,SHOME,doR1Serv_atHome,1;
ENDPROC

PROC rPowerON()
    rWZ_Home;
    rWZ_Maint;
    rWZ_Gripper;
    rWz_Clip;
ENDPROC
```

Todos los procesos de las zonas mundo, únicamente se pueden usar en rutinas de evento asociadas al evento PowerON,

2.10.7. Software de seguridad (SafeMove 2 Pro Virtual)

La mayoría de las instalaciones que integran robots industriales cuentan con programas de seguridad específicos, para disminuir los riesgos de colisión o atrapamiento, tanto de los operarios que podría encontrarse cerca del vallado exterior, como entre las distintas unidades mecánicas de la estación. La opción elegida por este sistema es el SafeMove 2 Pro Virtual, ya que ofrece más opciones de seguridad en la limitación de movimientos y también permite añadir otras restricciones mecánicas en el brazo robótico. Cuando se usa SafeMove, el controlador de seguridad garantiza un alto nivel de seguridad en el sistema del robot mediante el uso de funciones de supervisión que pueden detener al robot.

Al igual que toda la instalación de forma general, el módulo de seguridad y el SafeMove fueron diseñados cumpliendo principalmente las normas EN ISO 10218-1^[2] y 10218-2^[3], que son el estándar internacional de requisitos de seguridad y evaluación de riesgos de robots industriales. Pero, además, el sistema SafeMove cumple con otras normas de seguridad como la EN 60204-1^[5], referente a los requisitos generales de la seguridad de los equipos eléctricos. Para más información, consultar el anexo B1 donde se listan todas las normas aplicadas al sistema SafeMove.

Para introducir cambios en el sistema de SafeMove se debe conectar un ordenador al puerto de servicio del robot, entrar en la opción de SafeMove del RobotStudio y hacer allí los cambios deseados. Además, para realizar estos cambios se debe iniciar sesión en el sistema operativo del robot con el usuario de seguridad y sincronizar la estación del RobotStudio con la real.

Al entrar en detalle, en la aplicación descrita en este proyecto, el SafeMove se asegurará de que la celda se detenga cuando se presione una parada de emergencia o se abra la puerta. Además, se han creado vallas y herramientas virtuales para evitar la colisión entre el robot o las herramientas con las vallas de policarbonato. Estas vallas son una herramienta para que el controlador IRC 5 pueda percibir el entorno y limite sus movimientos para que no puedan atravesar el vallado virtual. Si la superficie

que forma el robot contando los volúmenes adicionales colisiona con la superficie que marca los límites de la zona de movimiento el robot se detiene y para volverlo a activar se debe pasar a manual y moverlo con el joystick para que vuelva a estar dentro de la zona. Este vallado presenta el inconveniente de que, si el robot está realizando un movimiento veloz y cuenta con un valor elevado de inercia, el tiempo de frenada en caso de atravesar la valla virtual será mayor, por lo tanto, podría ocurrir que siguiera avanzando indeseablemente y colisionará con el vallado real. Para prevenir este acontecimiento, el vallado virtual siempre se dota de unas medidas ligeramente inferiores al vallado real, así el límite del robot será anterior al límite físico.

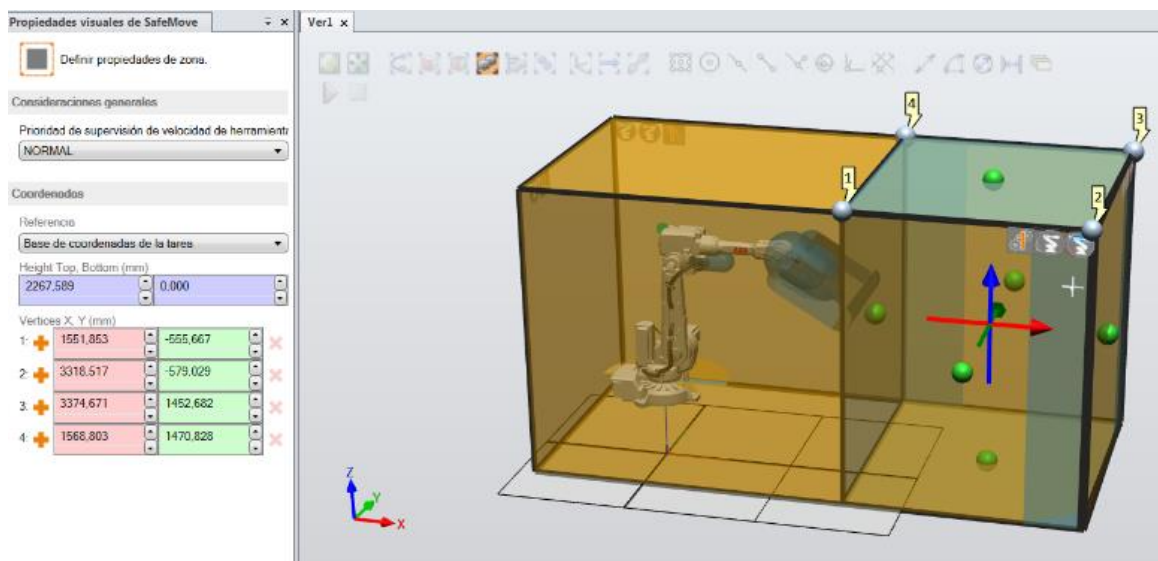


Figura 2.44. Captura del modelo 3D del vallado virtual (Derecha) y las medidas de esta área (Izquierda).

(Fuente: Vlad Zaharia)

Aunque el robot tiene unos interruptores que no permiten que se alcance el tope mecánico hay algunos casos que interesa limitar el valor que pueden adquirir. Como se puede observar en la Figura 2.45, en este caso se ha decidido reducir los límites del eje 1 a $[-35^\circ, 180^\circ]$ evitando así que el robot colisione indeseablemente con el vallado.

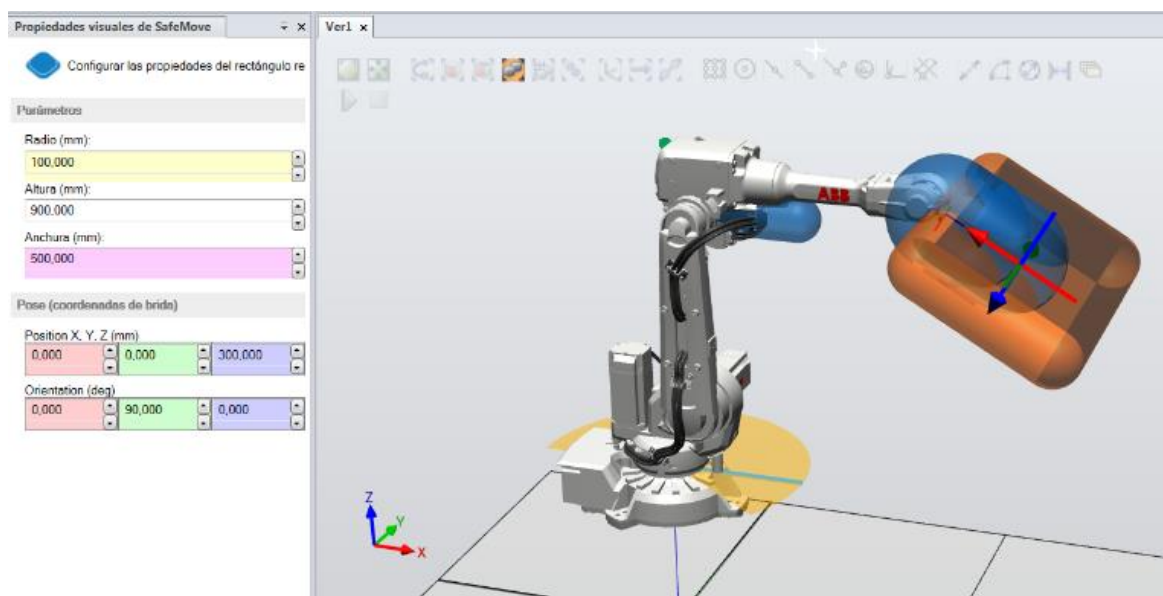


Figura 2.45. Captura del modelo 3D de las herramientas virtuales (Derecha) y las medidas de esta área (Izquierda). (Fuente: Vlad Zaharia)

El SafeMove también permite limitar la velocidad máxima del TCP en el modo de funcionamiento automático. En modo manual los robots de ABB tienen la velocidad limitada a 250 mm/s, pero en el modo automático esta velocidad únicamente se ve limitada por la velocidad introducida por el programa, o en caso de que esta velocidad sea muy elevada, se verá restringida por las características físicas y mecánicas del robot. La opción del SafeMove permite limitar la velocidad máxima del TCP tanto en los movimientos lineales como de reorientación y también la velocidad máxima de cada uno de los ejes sea cual sea el movimiento que esté realizando. En esta solución no se ha previsto el uso de la limitación de velocidad que ofrece SafeMove debido a que no se ha considerado necesario.

Aparte de estas acciones mencionadas el robot tiene activada también una comprobación periódica que la calibración del robot es correcta, con la finalidad de advertir y parar su funcionamiento en caso de que se pierda la calibración.

2.10.8. Interacción con periféricos

Como se ha visto en el caso de la rutina de limpieza de clips, el controlador del robot interactúa con los distintos periféricos, mediante la comunicación a través del PLC.

La dinámica de funcionamiento general de la celda se basa en la comunicación constante del PLC con el resto de los elementos de la celda, principalmente los controladores del robot y de la cámara, así como los elementos físicos tanto neumáticos como de seguridad.

La ejecución del programa está completamente supervisada por el PLC, por lo tanto, es este quien le indica en todo momento al robot que paso del programa debe ejecutar. Así pues, si se desea interactuar con algún periférico desde el IRC 5, como es el caso de abrir y cerrar la pinza de limpieza, este debe enviar una señal digital al PLC y el PLC es quien enviará la orden al actuador o periférico.

Lo mismo ocurre con la señal de disparo de la cámara. En primera instancia es el controlador del robot quien indica al PLC en que instante se desea realizar la captura, ya que el robot ha alcanzado la posición precisa para esto. Seguidamente el PLC valora si se cumplen las condiciones necesarias para realizar el disparo y en caso favorable, envía la señal de disparo al controlador CV-X150RP. Una vez realizada la captura, el controlador de la cámara envía los juicios de las herramientas de visión por PROFINET de vuelta al PLC, y este le comunica al IRC 5 cuales son los clips que se han colocado correctamente en la pieza, y cuáles no, para que en el momento de ejecutar la limpieza de la herramienta de clipaje, el robot sepa que dedos debe limpiar.

Esta comunicación se produce de igual manera con el resto de los periféricos y actuadores de la celda. Otros ejemplos son: el proceso de soltar la pieza junto a la herramienta de manipulación en el útil de clipaje, donde intervienen los cilindros que sustentan la herramienta; o el proceso de comunicación con la máquina de soldadura. Este último se explica en el apartado siguiente.

2.10.9. Comunicación con PLC y Máquina de soldadura

Una de las exigencias iniciales del proyecto fue la creación de una interfaz de comunicación entre el controlador IRC 5, el PLC S5-1515F y la máquina de soldadura. A raíz de esto, el autor de esta memoria elaboro un archivo en el que se detallan las señales necesarias que deben existir para la correcta comunicación entre los tres elementos, con el fin de cumplir con funcionamiento de ciclo de trabajo. Este documento se puede encontrar en el Anexo D1 de este trabajo: "Communication Interface".

Un punto imprescindible es que las seguridades, tanto del controlador del robot, del PLC como de la máquina de soldadura deben estar totalmente asociadas, asegurando así que, si se produce algún paro o error en alguno de los tres elementos, el resto actuará en consecuencia. Las señales de entrada y salida seguras se conectan a los circuitos de seguridad de la celda mediante una comunicación segura con un PLC de seguridad.

En el documento "Communication Interface", en primer lugar, se especifican las señales digitales de entrada y salida necesarias únicamente para la interacción del robot con la máquina de soldadura. Esta interacción se produce únicamente cuando el robot va a depositar la pieza acabada en la máquina de soldadura o recoge una pieza nueva. Así mismo, se detalla el significado de que las señales se encuentren a 1 o 0, y aparece un GRAFCET que resume las acciones y condiciones necesarias de este proceso. Este es asistido por breves explicaciones que facilitan su entendimiento a los lectores poco

familiarizados con el sistema. Para obtener más información sobre este tema, consultar el Anexo D1 “Communication Interface”.

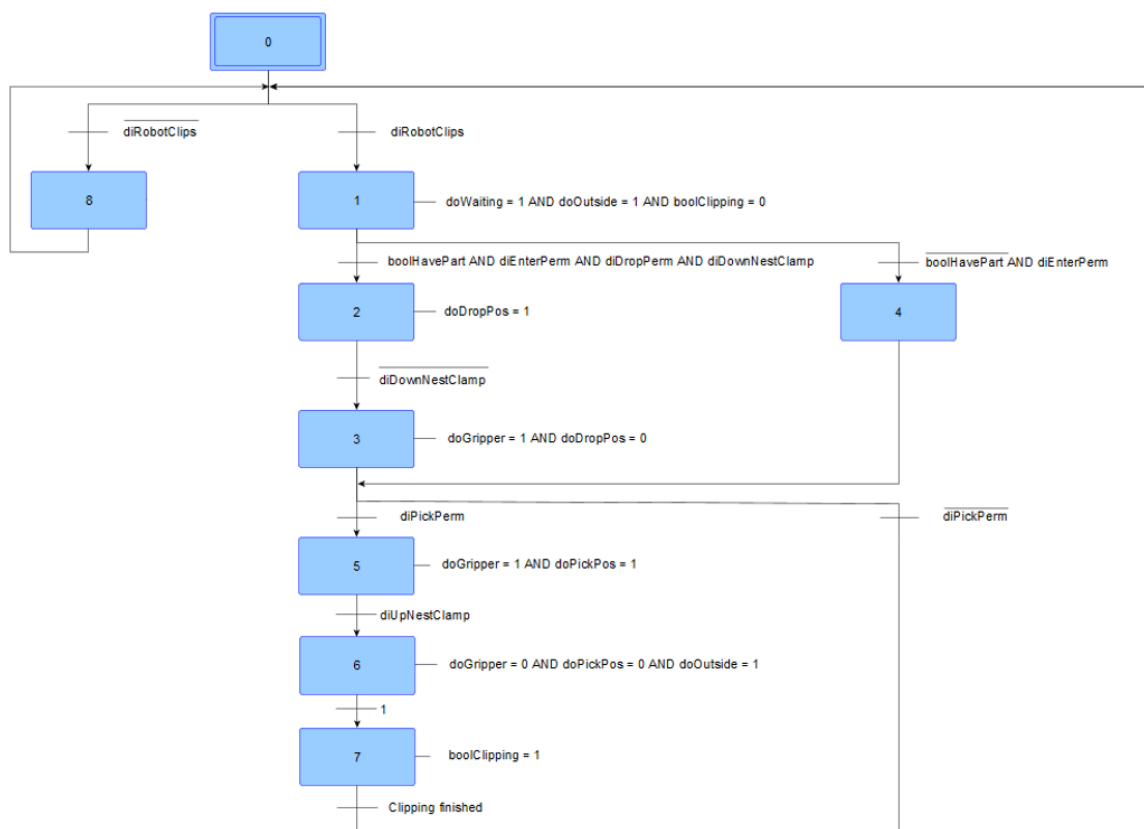


Figura 2.46. GRAFCET de los pasos más relevantes de la comunicación entre el IRC 5 y la máquina de soldadura.
(Fuente: Vlad Zaharia)

2.11. Programación del sistema de visión artificial

En lo que respecta a la visión artificial se ha programado una solución con el objetivo de reconocer la presencia de los clips plásticos en la pieza automovilística. Durante el proceso se realizan dos capturas de imagen, pero ambas cumplen las mismas condiciones de iluminación, entorno y posición relativa a la cámara respecto la pieza, asegurando así homogeneidad y repetitividad en las capturas, tanto al inicio como al final del ciclo.

El sistema de visión debe ser capaz de reconocer la presencia de la puerta en el utillaje, y después saber cuáles de los clips están ensamblados en la pieza. Al inicio es necesario identificar si la pieza se introduce en la máquina con algún clip ya colocado, para evitar que el robot colisione al intentar volver a colocar un clip en la misma posición. Al final del ciclo el propósito es comprobar que todos los clips se han colocado de forma correcta.

El modelo de cámara es KEYENCE CV-200M (Monocromático) con resolución de 2 Mpx y lente CA-LH8 con punto focal a 8 mm, controlada por el controlador CV-X150RP. El proceso seguido por el controlador CV-X150RP es: la adquisición, el procesamiento de las imágenes con el programa, análisis y posterior comprensión de los resultados enviados al PLC y que finalmente se muestran en la pantalla HMI.

A continuación, aparecen las características del sistema de visión artificial facilitados por el proveedor:

| | |
|--|--|
| Zona de visión | 900 x 700 |
| Resolución Cámara | 1600 x 1200 píxeles |
| Distancia cámara a pieza | 1000 mm |
| Resolución | 0,56 mm/píxel |
| Repetibilidad aplicación | 10 píxeles |
| Error debido a la variación de altura | $\pm 0,45 \%$ |
| Tolerancia o defecto mínimo a detectar | 6 mm <i>No se podrá detectar clips menores de 6mm ni encontrar su posición con menos de 6mm de diferencia respecto al patrón correcto.</i> |
| Tiempo de inspección | 250 ms aprox. |
| Posicionamiento | X: +/- 2 mm Y: +/- 2 mm Z: +/- 0 mm Ángulo: +/- 0° |

Figura 2.47. Resumen de las características del sistema de visión artificial (Fuente: Proveedor oficial KEYENCE)

La programación del sistema de visión artificial del presente proyecto se ha realizado de forma íntegra por el autor de este documento, directamente en el controlador CV-X150RP, con la ayuda de una pantalla conectada al controlador mediante VGA, y un ratón USB.

El primer paso para iniciar la programación de la visión artificial ha sido el ajuste correcto de la posición y la orientación de la cámara. La cámara está ubicada en un soporte fijo en la parte lateral de la celda para proporcionar una vista general de la pieza. El soporte permite el ajuste completo de la posición y la orientación de la cámara con el fin de ajustar su posición en caso de que esta interfiera en las trayectorias del robot. El panel luminoso situado en la parte superior de la celda asegura repetitividad en el brillo de las imágenes tomadas, independientemente de la luz ambiental presente. A continuación, se ajustó la apertura del diafragma para conseguir suficiente brillo en la imagen para poder distinguir los clips, que son de color blanco, de la pieza plástica de la puerta que es de color negro. Otra forma de ajustar la cantidad de luz de la imagen es modificar la velocidad de obturación. En este caso se ha seleccionado una velocidad de 2 ms. El tiempo de captura y posterior procesamiento de la imagen no tiene impacto en el tiempo de ciclo de la máquina ya que la captura se realiza al mismo tiempo que se mueve el robot. Seguidamente se procede a ajustar el enfoque de la óptica con la ayuda del propio programa, ya que este dispone de una opción de enfoque que indica el punto de mayor enfoque al ejecutar una captura de imágenes continua similar a un video.

El siguiente paso es configurar el controlador de forma general. Lo primero que se ha hecho es configurar su modo de arranque en Ejecución, para que cuando se encienda el controlador este inicie directamente en el modo de Ejecución del programa. Por lo contrario, el modo de configuración podría conllevar problemas a la hora de comunicar con el PLC, ya que la secuencia de envío de disparador y datos varía. También se debe activar la comunicación por PROFINET y dotar al controlador de una IP estática para que este comunique correctamente con el PLC.

Tras esto, se procede a crear el programa. En este proyecto son necesarios dos programas distintos para la visión, uno para cada modelo de pieza (Modelo1 y Modelo2). La gestión del cambio de programa se realiza en su totalidad desde el PLC ya que este puede enviar un comando a una dirección determinada del controlador CV-X150RP, y este interpretará el comando al instante. El PLC debe indicarle al controlador de la cámara que desea cambiar de programa, donde está ubicado el programa y qué programa desea ejecutar. Este proceso solamente se realiza cuando se vaya a cambiar la pieza de producción, hecho que ocurrirá raramente. La señal de disparo, es generada en primer lugar por el controlador del robot, cuando el robot se encuentre en la posición exacta en la que se desea tomar la fotografía, y es enviada al PLC. El PLC comprueba si se cumplen las condiciones necesarias para enviar la señal, ya que este dispone de más información de parte del controlador de la cámara, como por ejemplo el estado de los resultados de la captura anterior. Una vez se cumplan la totalidad de las

condiciones, el PLC es quien le envía la señal de Trigger al CV-X150RP. Entonces el controlador toma la imagen, la procesa y comunica los resultados de la presencia al PLC.

Una vez creados los programas, se crean las herramientas que se desea que tenga el programa. En primer lugar, se crea una herramienta de “Ajuste de posición por patrón de sombras”, con la finalidad de identificar un patrón de sombras e indicar si este ha sido desplazado o rotado respecto al patrón de entrenamiento. Esta herramienta es muy útil ya que las herramientas de presencia que se crearán a continuación estarán referenciadas a la posición de la herramienta de “Ajuste de posición por patrón de sombras”. Por lo tanto, aunque la puerta se haya desplazado por alguna razón en el utillaje, la visión por computadora será capaz de identificar la presencia de la pieza y de los clips. Esta misma herramienta también se usa para identificar la presencia de la pieza, ya que se ha establecido una coincidencia mínima de 80% con el patrón de entrenamiento para que esta herramienta de un OK. Por lo contrario, si la pieza no está colocada en su posición, el programa no identificará el patrón dentro del área de búsqueda y el juicio de la herramienta será NG.

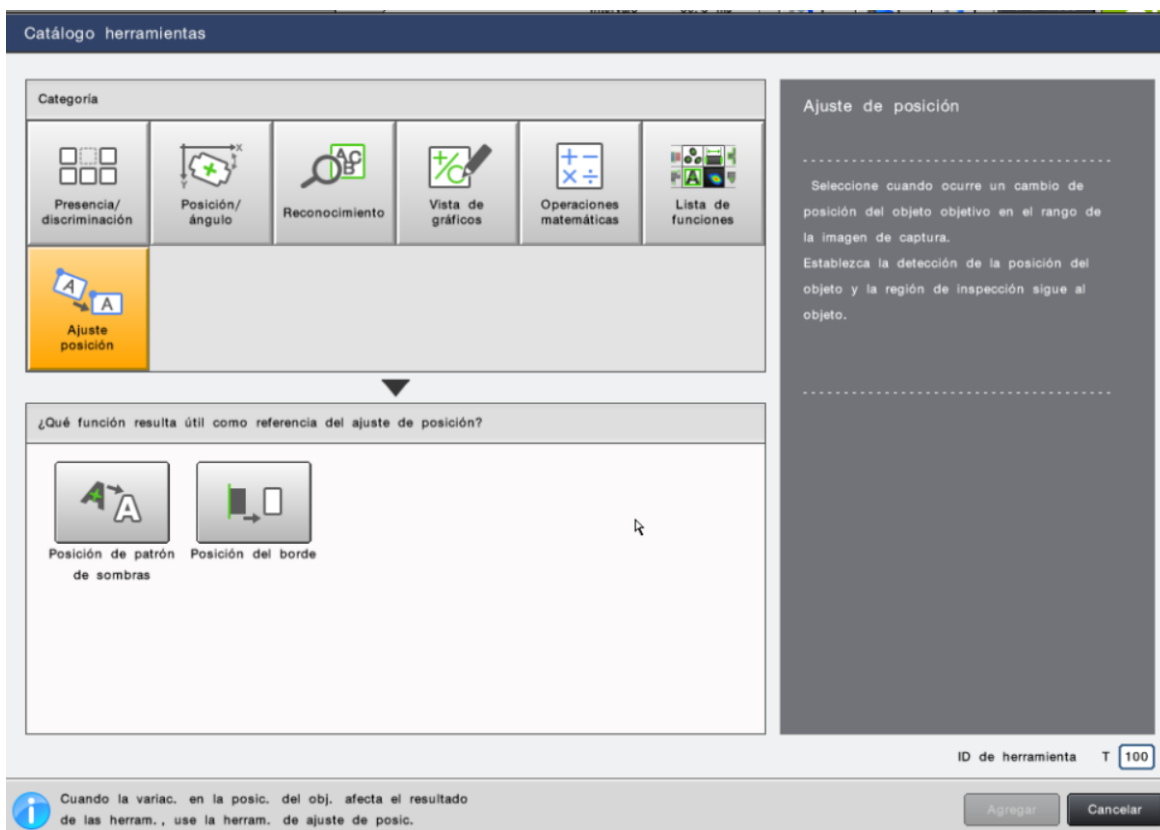


Figura 2.48. Menú de herramientas del controlador CV-X150RP (Fuente: Vlad Zaharia)

Como se ha comentado, a continuación, se crean las herramientas de detección de presencia de los clips. Harán falta 8 herramientas, una para cada clip. Al existir mucho contraste entre el color negro de

la pieza y el color blanco del clip, en primera instancia se decidió usar la herramienta de detección de Área Blanca/Negra, en la que la imagen es binarizada de forma automática o manual (en caso de que se desee modificar el valor umbral), y posteriormente se cuentan el número de píxeles blancos o negros (en este caso blancos) existentes en un área determinada. De esta forma, se le indica a la herramienta el valor mínimo y máximo de píxeles blancos que debe haber para que el juicio sea positivo. En todos los clips se ha establecido un valor mínimo de 50 píxeles y un valor máximo de 350 píxeles. Ha sido necesario ajustar el valor umbral en cada una de las herramientas debido a que la superficie de la pieza tiene una textura brillante, y con el aumento de incidencia de luz en ella, la herramienta podría interpretar la superficie brillante como píxeles blancos.

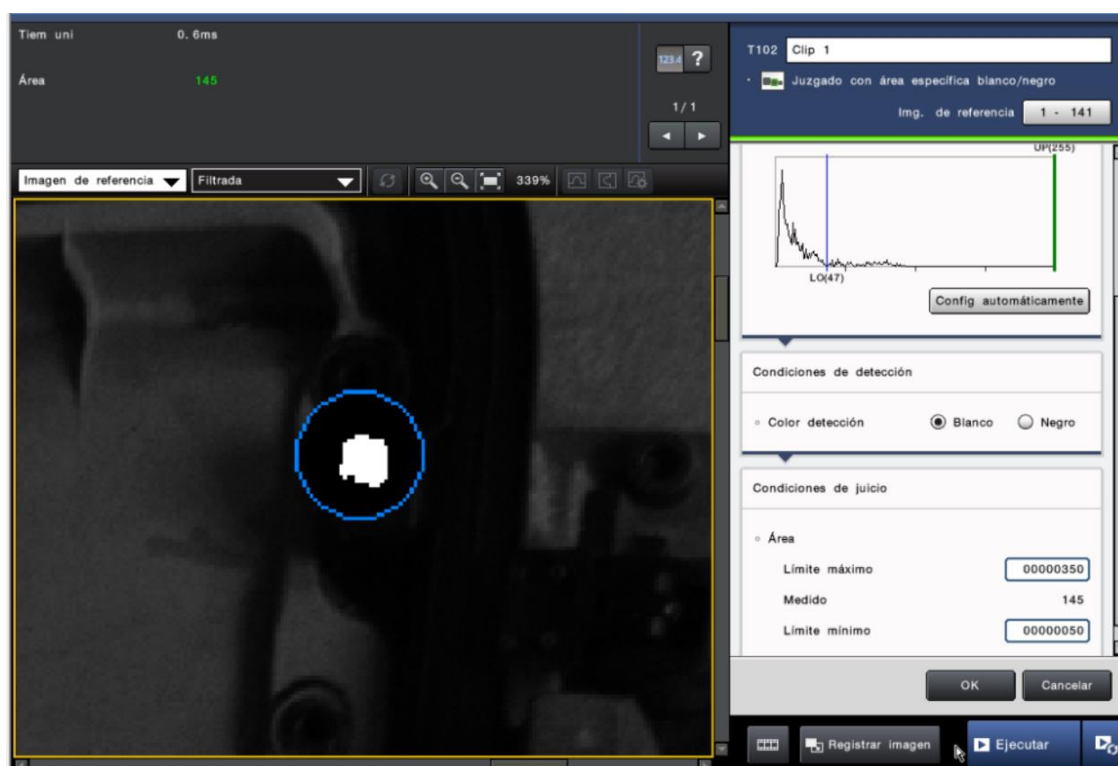


Figura 2.49. Configuración de la herramienta de detección de presencia por área blanca (Fuente: Vlad Zaharia)

Este tipo de herramienta presenta muy buenos resultados en siete de los ocho clips. El problema aparece cuando la parte en la que se introduce el clip deja de ser negra, como es el caso del clip número 2, donde la pieza tiene una tonalidad de gris claro. Para solucionar este problema se decidió crear una herramienta específica para este clip y dejar el resto con la herramienta de detección de área blanca.

Para el clip número 2 se creó una herramienta de “Detección de patrón de sombras”, en la que se entrena un patrón y se especifica el área de búsqueda para este. El valor mínimo de coincidencia necesario para un resultado OK ha sido establecido en un 70% ya que el área de búsqueda es muy ajustada y difícilmente podrá dar falsos OK. Se ha querido dejar un margen de 30% para aportar más

flexibilidad al programa, ya que al ser una celda sin techo la luz podría variar en algunas condiciones y el patrón de sombras sería ligeramente distinto.

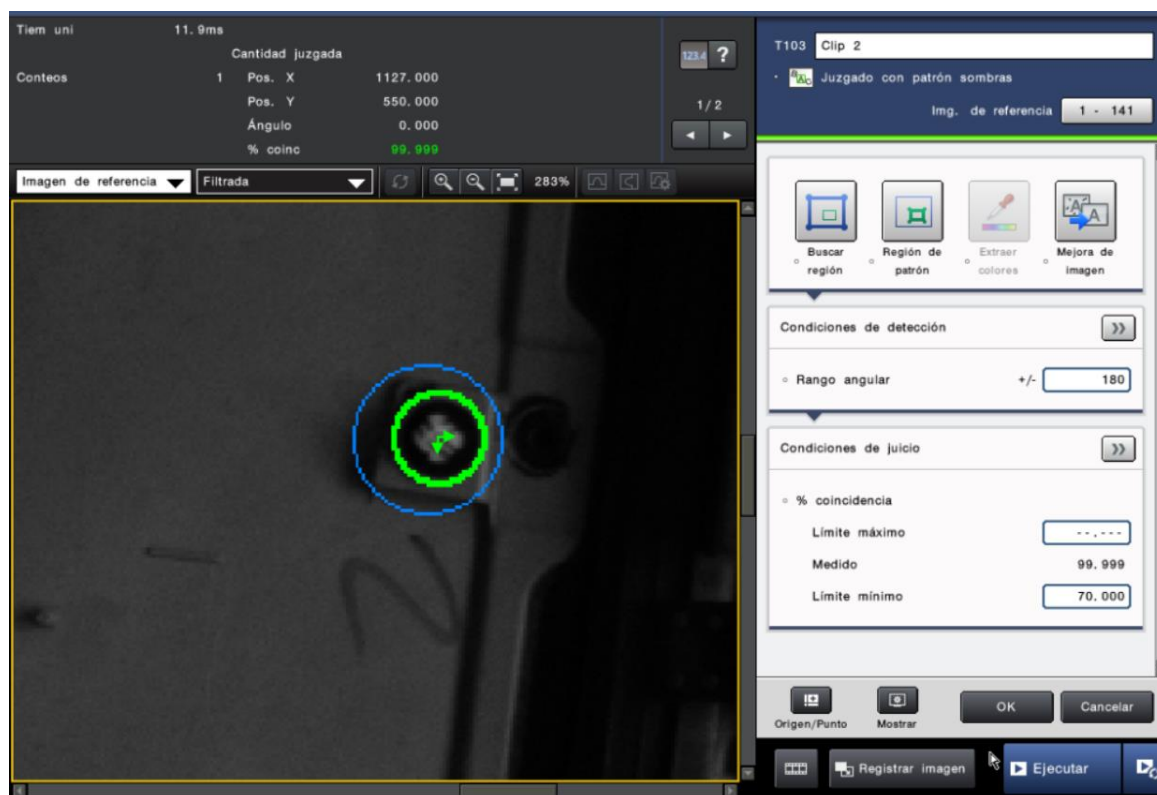


Figura 2.50. Configuración de la herramienta de detección de presencia por patrón de sombras. (Fuente: Vlad Zaharia)

Tras crear las herramientas de juicio, se debe de seleccionar las señales que se envían al PLC por PROFINET, y cuál es el juicio de cada una de ellas. En este caso se decidió enviar en primer lugar un bit que indique la presencia de la pieza, y a continuación un byte con el juicio de las 8 herramientas.

Lista de datos de envío de PROFINET

Área de asignación de bits

| Dirección | Valor | Descripciones |
|-----------|-------------------------------------|-------------------|
| 0003:bit2 | <input type="checkbox"/> | (Reservado) |
| 0003:bit3 | <input checked="" type="checkbox"/> | RUN |
| 0003:bit4 | <input type="checkbox"/> | (Reservado) |
| 0003:bit5 | <input type="checkbox"/> | (Reservado) |
| 0003:bit6 | <input type="checkbox"/> | (Reservado) |
| 0003:bit7 | <input type="checkbox"/> | (Reservado) |
| 0004:bit0 | <input checked="" type="checkbox"/> | Tool Judge Value0 |
| 0004:bit1 | <input checked="" type="checkbox"/> | Tool Judge Value1 |
| 0004:bit2 | <input checked="" type="checkbox"/> | Tool Judge Value2 |
| 0004:bit3 | <input checked="" type="checkbox"/> | Tool Judge Value3 |
| 0004:bit4 | <input checked="" type="checkbox"/> | Tool Judge Value4 |
| 0004:bit5 | <input checked="" type="checkbox"/> | Tool Judge Value5 |
| 0004:bit6 | <input checked="" type="checkbox"/> | Tool Judge Value6 |
| 0004:bit7 | <input checked="" type="checkbox"/> | Tool Judge Value7 |
| 0005:bit0 | <input checked="" type="checkbox"/> | Tool Judge Value8 |
| 0005:bit1 | <input type="checkbox"/> | Tool Judge Value9 |

Área de asignación de bytes

| Dirección | Valor | Descripciones |
|-----------|----------|----------------|
| 0012 | 0000 | Error Code |
| 0014 | 0000 | (Reservado) |
| 0016 | 00000009 | Total Count |
| 0020 | 00000000 | Command Result |
| 0024 | 00000000 | Command Data1 |
| 0028 | 00000000 | Command Data2 |
| 0032 | 00000000 | Command Data3 |
| 0036 | 00000000 | Command Data4 |
| 0040 | 00000000 | Command Data5 |
| 0044 | 00000000 | Command Data6 |
| 0048 | 00000000 | Result Data1 |
| 0052 | 00000000 | Result Data2 |
| 0056 | 00000000 | Result Data3 |
| 0060 | 00000000 | Result Data4 |
| 0064 | 00000000 | Result Data5 |
| 0068 | 00000000 | Result Data6 |

☐ Salida manual

Comunicación: Habilitado Estado de comunicación cíclica: **Conectado** **Cerrar**

Figura 2.51. Señales de envío del controlador CV-X150RP al PLC mediante PROFINET en caso de juicio OK de todas las herramientas. (Fuente: Vlad Zaharia)

Al finalizar el proceso de programación se cambia el modo de funcionamiento del controlador al Modo de Ejecución. Cabe mencionar que todas las herramienta se ejecutan al mismo tiempo ya que unicamente se realiza una captura por inspección y el mismo programa sirve para ambas inspecciones.



Figura 2.52. Captura de la pantalla de ejecución del controlador CV-X150RP donde se observan los juicios de las herramientas. (Fuente: Vlad Zaharia)

2.12. Modificaciones realizadas en el diseño

Durante el transcurso del proyecto se han ido efectuando modificaciones tanto en el diseño de la celda como en las características de los programas, con el fin de mejorar o añadir funcionalidades nuevas que anteriormente no se tenían previstas. Estas implementaciones han aparecido tanto por parte del equipo de proyecto de ABB como de los deseos del cliente. La dinámica de reuniones continuas que ha favorecido la comunicación ha sido un factor clave para las mejoras que se han realizado.

La primera modificación del diseño que se implementó fue la de cambiar la posición del utillaje, ya que, en la propuesta inicial, esta era horizontal y se encontraba entre la plataforma del robot y la máquina de soldadura. Este utillaje se sustituyó por uno inclinado, prácticamente vertical, situado en la parte trasera de la celda, permitiendo así el acceso al carro de mantenimiento del utillaje. Este cambio también significó rediseñar el utillaje ya que este debía ser capaz de mantener la pieza firme y que la pieza no pueda caer debido a la fuerza de la gravedad. La decisión de realizar esta modificación fue acordada durante una reunión con el cliente y el equipo del proyecto.

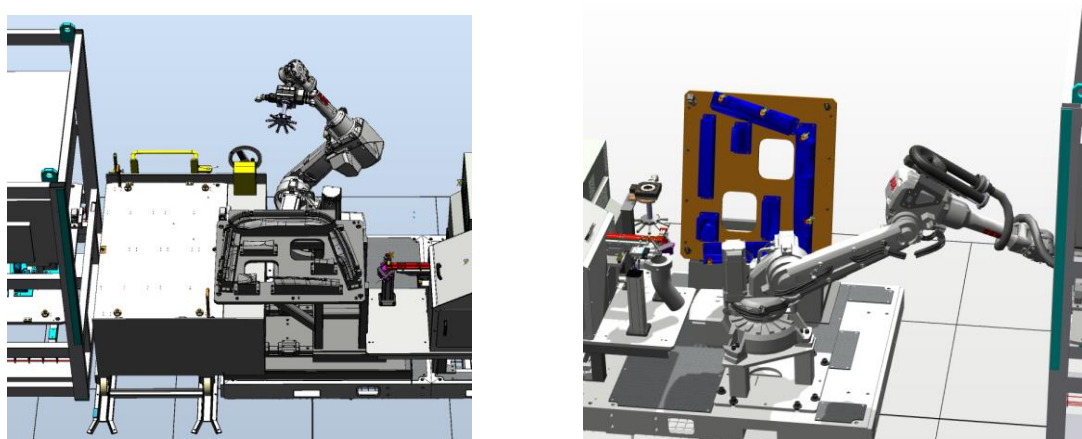


Figura 2.53. y Figura 2.54. Comparación entre la posición inicial (Izquierda) y final (Derecha) del utillaje. (Fuente: Vlad Zaharia)

La segunda modificación fue la de cambiar totalmente la posición de la cámara suponiendo un diseño nuevo para el soporte de esta. Durante una de las primeras visitas del cliente, el técnico de visión, autor de este proyecto, propuso que se modificara la posición de la cámara, situándola en un soporte fijo en frente del utillaje. También indicó claramente en que instante se podían realizar ambas capturas ya que conocía de primera mano el ciclo de la máquina. Esto fue así porque estuvo presente en la etapa de diseño de la celda, participó en la creación de la primera simulación (aunque posteriormente realizó por completo la simulación explicada en este documento), y además realizó la interfaz de

comunicación necesaria entre el IRC 5, el PLC y la máquina de soldadura, previamente a la programación de los controladores de la cámara y el robot.

Tras analizar debidamente su propuesta, el cliente la aceptó e inició el proceso de diseño del soporte de la cámara, en el cual también participó. Este soporte está hecho por tres perfiles de aluminio de 40x40 mm colocados ortogonalmente entre sí y sujetos por escuadras de aluminio. De esta forma existe la posibilidad de desplazar la cámara tridimensionalmente para permitir que el robot se mueva más libremente, ya que esta se sitúa justo encima de él. La pieza en la que se atornilla la cámara permite modificar su posición angular, dando libertad al técnico de visión para su total ajuste.



Figura 2.55 y Figura 2.56. Captura del modelo 3D (Izquierda) y fotografía (Derecha) del soporte de la cámara.
(Fuente: Vlad Zaharia)

Un cambio que se tuvo que realizar después del montaje de la celda fue el de cambiar el tamaño y la posición del soporte (raqueta) del paquete energético situado en el eje 6 del robot, ya que inicialmente este fue colocado muy cerca del eje 5 del robot, limitando mucho los movimientos de reorientación. Esta modificación fue solicitada por el equipo de programación del robot, del cual también forma parte el autor de esta memoria.



Figura 2.57. Fotografía del soporte (raqueta) del paquete energético situado en el eje 6 del robot. (Fuente: Vlad Zaharia)

Una vez la celda se encontraba en una fase de desarrollo avanzada, y ya se podían realizar ciclos con pieza, se decidió cambiar el material de los dedos de las pinzas de la herramienta de manipulación. Estos dedos inicialmente eran de aluminio, pero el cliente solicitó que en su extremo tuvieran un recubrimiento de goma para evitar que el aluminio dañe el exterior de la puerta, ya que es la parte visible para el cliente final. El cambio de material supuso el rediseño de todos los dedos ya que no era posible añadir más grosor a las pinzas que ya estaban instaladas.

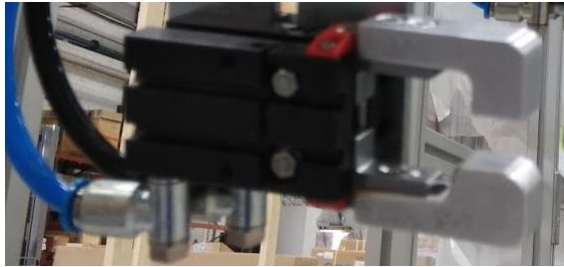


Figura 2.58. y Figura 2.59. Fotografía de una pinza de la herramienta de manipulación (Izquierda) y fotografía de un dedo de goma junto a los dedos de aluminio (Derecha). (Fuente: Vlad Zaharia)

La última modificación que se llevó a cabo fue la de introducir los cerrojos en el interior del perfil de las puertas. Esta modificación se realizó con el objetivo de reducir aún más el ancho de la celda.

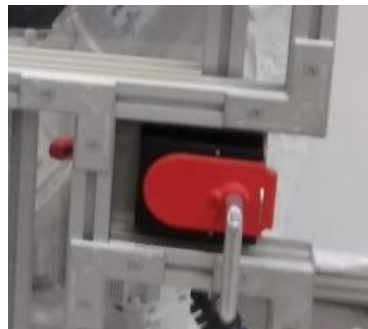


Figura 2.60. y Figura 2.61. Fotografía del cerrojo visto desde el exterior (Izquierda) y desde el interior (Derecha). (Fuente: Vlad Zaharia)

2.13. Comparación del tiempo de ciclo y optimización del proceso

Teniendo en cuenta que el tiempo de ciclo resultante de la simulación es de 58 segundo y se sitúa por debajo del requisito del cliente, de 62 segundo, la optimización del proceso no ha sido prioritaria durante la etapa de programación del robot. Esto se debe a que esta fue debidamente realizada durante la creación de la simulación. Aun así, se intentó disminuir el tiempo de ciclo, teniendo especial cuidado en que estos cambios no supusieran un peligro en los elementos mecánicos de la celda.

En primera instancia, lo que se suele hacer para disminuir el tiempo de ciclo es aumentar la velocidad de movimiento en las trayectorias que esto sea posible. Esto puede suponer mayor carga para los motores de los ejes y los frenos de estos, disminuyendo significativamente su tiempo de vida. Por lo tanto, es el cliente que debe decidir si desea realizar este cambio o no. En este proyecto, teniendo en cuenta que los resultados obtenidos ya se situaban por debajo del tiempo de ciclo requerido, se decidió que solamente se aumentarían las velocidades de las trayectorias que no supongan un desgaste de los elementos mecánicos.

Un factor que por lo contrario causa el aumento de tiempo de ciclo respecto la simulación es la longitud del soporte (raqueta) del paquete energético del robot. Es un hecho que no se contempló en la simulación ya que este soporte puede o no ser finalmente necesario, dependiendo de las características y el peso del paquete energético seleccionado. El hecho de no haber tenido en cuenta este soporte en la simulación permite colocar los clips con el orden de dedos consecutivo, en cambio en la realidad, para evitar que la raqueta colisione con la pieza se ha tenido que modificar el orden de algunos de los dedos que se utilizan para clipar la pieza, aunque el orden de clipaje de los clips se mantiene como el de la simulación. Esto produce algunos giros durante el proceso de clipaje que no se habían previsto, por lo tanto, resultaban en un ligero aumento del tiempo de ciclo.

A pesar de esto, al finalizar la programación del robot se ha obtenido un tiempo de ciclo de 53 segundos, realizando la limpieza de todos los dedos al igual que en la simulación. Si sólo se tiene en cuenta el ciclo habitual de la máquina en el que no deberá limpiar ninguno de los dedos, se alcanza un tiempo de ciclo de 44 segundos, estando muy por debajo del tiempo solicitado.

CAPÍTULO III

Resultados

En este tercer capítulo se expone el resultado final del proyecto y como este ha sido concluido.

3.1. Aprobación del cliente

Al finalizar el ensamblaje de todos los elementos físicos nombrados en el Capítulo II, y realizar la pertinente programación del controlador del robot, de la cámara y del PLC, se completó la solución final del proyecto.

Antes de presentar la solución final, se realizan unas pruebas internas, con el fin de asegurar que la inspección del cliente se llevará a cabo exitosamente. Con este fin se decide subcontractar a una empresa experta en análisis de seguridades en instalaciones industriales, para que esta indique si la celda cumple con las normativas aplicadas, o por lo contrario se debe realizar alguna modificación que solvente la existencia de algún riesgo. En este caso, la inspección concluyo con un resultado positivo, debido a que el equipo de proyectos de ABB ya está familiarizado con este tipo de instalación y previo la implementación de elementos de seguridad.

Por otra parte, se realizaron pruebas de producción, en los que se sometieron a ciclos de estrés todos los procesos de la celda, con la finalidad de detectar fallos poco habituales, pero que también tuvieron ser corregidos. En este aspecto, se modificaron algunos puntos de la trayectoria de robot para ajustarse aún más a la posición deseada. También se modificaron algunos fragmentos de programa en los que se observó que no se ejecutaban las rutinas de forma esperada.

Finalmente, es el cliente quien realiza una inspección exhaustiva para comprobar que la instalación puede cumplir exitosamente con todas las funcionalidades demandadas. Otro aspecto en el que el cliente pone especial hincapié es en el cumplimiento de la normativa acordada, así como, se analizan los riesgos y seguridades de la celda.

Tras la inspección, el cliente realiza un estudio de los resultados obtenidos y posteriormente comunica al Project Manager la decisión de iniciar el proceso de desarme de la celda para su transporte. Si se produjera el caso contrario, en el que la respuesta fuera negativa, se tendría que modificar algún elemento o implementar alguna funcionalidad extra en el menor tiempo posible, para no realizar la entrega fuera del límite de tiempo establecido.

3.2. Resultado final del proyecto

El resultado obtenido al finalizar el proyecto fueron cuatro celdas robotizadas, de características idénticas que cumplen la función de ensamblar clips dentro de una pieza de plástico de mayor tamaño. Cumpliendo con los requisitos iniciales del cliente, la solución integra un robot articulado IRB 2600 y el controlador CV-X150RP junto con la cámara CV-200M. Además de realizar el ciclo habitual de producción la celda cuenta con funcionalidades extra que garantizan versatilidad en su funcionamiento, como es el caso de la rutina de recuperación del robot en caso de un paro inesperado en medio del ciclo de producción.

Durante todo el proceso se produjeron cambios tanto en el diseño de la celda como en algunas funcionalidades con el fin de satisfacer las necesidades que fueron surgiendo durante el desarrollo de este.

Por último, tras pasar la aprobación final del cliente se realiza la entrega de las celdas a la planta de producción situada en Francia. Allí se realiza un ajuste final de algunas partes mecánicas que podrían haber modificado ligeramente su posición durante el proceso de transporte. También se debe de comprobar el correcto funcionamiento del sistema de visión artificial, porque la celda no dispone de techo y la luz incidente varía según las condiciones de luz propias de la planta de fabricación. Además, se imparte una formación a los trabajadores que deban interactuar con la instalación con el fin de transmitir tanto las normas de seguridad y mantenimiento que estos deben seguir, como para detallar las funcionalidades de la celda y los pasos necesarios para llevar a cabo la producción de forma satisfactoria. Esta formación es impartida por la propia empresa cliente del proyecto.

A continuación, se muestra una tabla que resume las aportaciones o fases del proyecto que fueron realizadas o en las que ha colaborado el autor de esta memoria:

| Número | Aportación |
|--------|--|
| 1 | Colaboración en la fase de diseño |
| 2 | Colaboración en la gestión y la toma de decisiones del proyecto |
| 3 | Creación de la estación virtual y la simulación 3D |
| 4 | Creación de la interfaz de comunicación entre el controlador IRC5 y la máquina de soldadura (Anexo D1) |
| 5 | Colaboración en la programación del controlador del robot (IRC 5) |
| 6 | Ajuste y programación del sistema de visión artificial |
| 7 | Elaboración de la documentación técnica de la instalación (Anexo A1 "User Manual") |
| 8 | Elaboración de la guía de estandarización de la programación |
| 9 | Elaboración de la Guía rápida de simulación (Anexo C1) |
| 10 | Elaboración de la Guía rápida de programación (Anexo C2) |

Tabla 3.1. Resumen de las aportaciones más relevantes del autor al proyecto (Fuente: Vlad Zaharia)

Presupuesto y análisis económico

El presupuesto contiene los gastos previstos para la ejecución del proyecto. En la siguiente tabla se muestran los diferentes conceptos de los gastos del proyecto y su valor económico. Los números que aparecen son aproximaciones ya que por motivos de confidencialidad no se pueden mostrar los valores reales.

| DESCRIPCIÓN | PRECIO |
|---|--------------------|
| Robot IRB2600-20/1.65 + Controlador IRC 5 | 20.000,00 € |
| Fabricación plataformas | 600,00 € |
| Vallado | 2.400,00 € |
| Sistema de cambio de herramienta | 8.000,00 € |
| Alimentador de clips plásticos | 19.000,00 € |
| Diseño piezas mecánicas | 4.000,00 € |
| Fabricación de elementos mecánicas | 19.000,00 € |
| Material neumático | 2400,00 € |
| Material eléctrico | 1550,00 € |
| Sistema de visión | 4.000,00 € |
| Sensores | 700,00 € |
| Inspección de seguridad | 2.000,00 € |
| TOTAL | 84.650,00 € |

Tabla 4.1. Tabla de los costes materiales de una celda robotizada del presente proyecto. (Fuente: Vlad Zaharia)

| CARGO | NÚMERO x HORAS | PRECIO | TOTAL |
|---|----------------|---------|--------------------|
| Jefe de proyecto (Project Manager) | 400 | 20,00 € | 8.000,00 € |
| Técnicos de programación robot y visión | 2 x 400 | 12,50 € | 10.000,50 € |
| Técnico de programación PLC | 400 | 15,00 € | 6.000,00 € |
| Técnicos de simulación | 2 x 280 | 15,00 € | 7.000,50 € |
| Mecánicos | 2 x 160 | 10,00 € | 3.200,00 € |
| Electricistas | 2 x 160 | 10,00 € | 3.200,00 € |
| TOTAL | | | 37.000,00 € |

Tabla 4.2. Tabla de los gastos de trabajo del proyecto. (Fuente: Vlad Zaharia)

Tras analizar los costes económicos, tanto del material que compone la celda como de las horas de trabajo de los miembros de equipo, se ha determinado que el coste total de una celda robotizada ha sido de 121.650,00 €, excluyendo el coste de la máquina de soldadura, ya que esta es propiedad del cliente.

Análisis del impacto ambiental

La celda robotizada se ubicará dentro de una nave industrial, así que esta no provocará ningún impacto visual o paisajístico.

Analizando los niveles de ruido ambiental de la celda descrita en este documento, se puede afirmar que estos no exceden los 69 dB de ruido. A continuación, se muestra una tabla con las especificaciones del robot IRB 2600:

| Datos | Descripción | Nota |
|--------------------------------------|--|--|
| Nivel de ruido propagado por el aire | Nivel de presión sonora en el exterior del área de trabajo | < 69 dB (A) Leq (de acuerdo con la Directiva de máquinas 2006/42/CE) |

Tabla 4.3. Tabla de las especificaciones del robot IRB 2600 sobre sus niveles de ruido ambiental. (Fuente: Datos técnicos IRB 2600)^[14]

Si el equipo se debe desarmar al final de su vida útil, todos los componentes se deben desarmar y separar de acuerdo con el tipo de material para permitir el reciclaje. La mayoría de los componentes y la estructura están hechos de acero y, por lo tanto, se pueden desarmar y reciclar como chatarra. Todos los aceites y grasas usados, así como las baterías gastadas, deben desecharse de acuerdo con la legislación vigente del país en el que se instala la celda.

En lo que respecta a la huella de CO₂ del proyecto, esta principalmente es causada por el dióxido de carbono emitido al ambiente durante la etapa de transporte. Se debe considerar la distribución de los diferentes elementos desde el origen hasta la nave de Sant Quirze del Vallés y el transporte de la celda una vez finalizada, hasta la nave de destino en Francia. Estos transportes se realizan con el camión como medio de transporte por carretera.

Conclusiones

En el presente trabajo se han descrito los pasos a seguir para desarrollar una celda robotizada para una aplicación en el sector automovilístico. La función principal de la instalación es la de ensamblar clips en el interior de una pieza de plástico de mayor tamaño, que pertenece al interior de una puerta de automóvil. Para llevar a cabo esta acción se ha optado por el uso de un robot articulado industrial IRB 2600, que manipula la pieza de automóvil e introduce clips en ella.

En primer lugar, es importante remarcar que se han cumplido todos los objetivos propuestos al inicio del proyecto. Al final de este proceso se ha conseguido crear una celda robotizada que cumple todas las especificaciones del cliente.

Las partes del proyecto, que más relevancia se les atribuye en este documento, son el estudio previo de la solución, la creación de la simulación 3D, la programación del controlador del robot y la programación del sistema de visión artificial.

La creación de la estación virtual, y posteriormente de la simulación de los movimientos y el ciclo de producción, supone un gran ahorro económico para el proyecto, ya que este permite prever los inconvenientes que se pudieran producir, además de realizar cambios en el diseño con suma facilidad y optimizar al máximo el tiempo de ciclo.

La adecuada gestión de proyecto y la constante comunicación con el cliente ha favorecido el desarrollo de este, ya que ha existido una constante retroalimentación de las decisiones tomadas en el transcurso del proyecto. Hay que destacar también que la planificación propuesta inicialmente se pudo llevar a cabo correctamente, derivando así a que el proyecto fuera un éxito y que los clientes estuvieran satisfechos con el producto final.

También se ha tenido en cuenta durante todas las etapas del desarrollo el cumplimiento de la normativa europea que acata este tipo de instalaciones, con el fin de garantizar en todo momento la seguridad de los técnicos y operarios, así como la integridad de los elementos de la celda.

La realización de este proyecto ha permitido poner en práctica los conocimientos adquiridos durante la estancia como becario en ABB, aparte de los conocimientos adquiridos durante la carrera, siendo la asignatura de Robótica Industrial y Visión por Computadora, entre otras, esencial para el desarrollo de este. Esta experiencia ha resultado ser muy satisfactoria para el autor del trabajo, ya que este ha podido participar en gran parte de las fases del proyecto. El trabajo ha supuesto una mejora personal en las áreas de trabajo mencionadas.

Es importante comentar que la puesta en marcha de la instalación en la propia planta del cliente fue satisfactoria, y esta se encuentra actualmente en producción.

Otra conclusión que se puede obtener de este proyecto es que el hecho de utilizar robots en las celdas industriales es una ventaja competitiva muy grande, por su rentabilidad económica, mejora y aumento de la producción. Por estas razones se prevé que, en los próximos años, la robótica será imprescindible en la producción industrial de cualquier sector.

El presente documento también tiene la función final de servir como guía para futuros proyectos de características similares. Y sería una buena mejora para proyectos de las mismas características usar la estandarización propuesta por el autor de esta memoria, con el fin de agilizar los procesos de programación.

Referencias bibliográficas

Normas:

- [1] International Organization for Standardization. *ISO 8373:2012: Robots and robotic devices – Vocabulary* [en línea]. 2012 [Consulta: 23 abril 2019]. Disponible a: <<https://www.iso.org/obp/ui/es/#iso:std:iso:8373:ed-2:v1:en>>
- [2] Asociación Española de Normalización y Certificación. *UNE-EN ISO 10218-1:2012: Robots y dispositivos robóticos. Requisitos de seguridad para robots industriales. Parte 1: Robots* [en línea]. 2012 [Consulta: 28 abril 2019]. Disponible a: <<https://www.une.org/encuentra-tu-norma/busca-tu-norma/norma?c=N0049289>>
- [3] Asociación Española de Normalización y Certificación. *UNE-EN ISO 10218-2:2011: Robots y dispositivos robóticos. Requisitos de seguridad para robots industriales. Parte 2: Sistemas robot e integración.* [en línea]. 2011 [Consulta: 28 abril 2019]. Disponible a: <<https://www.une.org/encuentra-tu-norma/busca-tu-norma/norma?c=N0048668>>
- [4] Asociación Española de Normalización y Certificación. *UNE-EN ISO 12100:2012: Seguridad de las máquinas. Principios generales para el diseño. Evaluación del riesgo y reducción del riesgo.* [en línea]. 2012 [Consulta: 28 abril 2019]. Disponible a: <<https://www.aenor.com/normas-y-libros/buscador-de-normas/une?c=N0049405>>
- [5] Asociación Española de Normalización y Certificación. *UNE-EN 60204-1:2007: Seguridad de las máquinas. Equipo eléctrico de las máquinas. Parte 1: Requisitos generales.* [en línea]. 2007 [Consulta: 28 abril 2019]. Disponible a: <<https://www.une.org/encuentra-tu-norma/busca-tu-norma/norma?c=N0038545>>
- [6] Asociación Española de Normalización y Certificación. *UNE-EN 614-1:2006: Seguridad de las máquinas. Principios de diseño ergonómico. Parte 1: Terminología y principios generales.* [en línea]. 2008 [Consulta: 28 abril 2019]. Disponible a: <<https://www.une.org/encuentra-tu-norma/busca-tu-norma/norma/?c=N0043852>>
- [7] Asociación Española de Normalización y Certificación. *UNE-EN 574:1997: Seguridad de las máquinas. Dispositivos de mando a dos manos. Aspectos funcionales. Principios para el diseño.* [en línea]. 2008 [Consulta: 28 abril 2019]. Disponible a: <<https://www.aenor.com/normas-y-libros/buscador-de-normas/une/?c=N0042162>>
- [8] Asociación Española de Normalización y Certificación. *UNE-EN ISO 14120:2016: Seguridad de las máquinas. Resguardos. Requisitos generales para el diseño y construcción*

de resguardos fijos y móviles. [en línea]. 2016 [Consulta: 28 abril 2019]. Disponible a: <https://www.une.org/encuentra-tu-norma/busca-tu-norma/norma?c=N0056446>>

[9] Asociación Española de Normalización y Certificación. *UNE-EN ISO 13849-1:2016: Seguridad de las máquinas. Partes de los sistemas de mando relativas a la seguridad. Parte 1: Principios generales para el diseño*. [en línea]. 2016 [Consulta: 28 abril 2019]. Disponible a: <https://www.une.org/encuentra-tu-norma/busca-tu-norma/norma?c=N0057029>>

Manuales técnicos:

[10] ABB. *Descripción general de RAPID*. ABB, 2018 [Consulta: 5 mayo 2019].

[12] ABB. *Especificaciones del producto. Controlador IRC5*. ABB, 2018 [Consulta: 5 mayo 2019].

[13] ABB. *IRC5. Descripción del entorno*. ABB, 2018 [Consulta: 5 mayo 2019].

[14] ABB. *Especificaciones del producto. IRB 2600*. ABB, 2018 [Consulta: 5 mayo 2019].

[15] KEYENCE. *Serie CV-X100: Manual de instrucciones del controlador. MCV27MX*, 2018 [Consulta: 27 abril 2019].

Guías:

[16] Ministerio de Industria, Energía y Turismo. *Sistemas de automatización y robótica para las pymes españolas*. [en línea]. Gobierno de España. 2015 [Consulta: 14 mayo 2019]. Disponible a: <https://www.eoi.es/es/savia/publicaciones/21128/sistemas-de-automatizacion-y-robotica-para-las-pymes-espanolas>>

[17] Iñigo Carrión Rosende, Iosune Berasategi Vitoria. *Guía para la elaboración de proyectos*. [en línea]. Instituto vasco de cualificaciones y formación profesional. 2010 [Consulta: 8 febrero 2019]. Disponible a: https://www.pluralismoyconvivencia.es/upload/19/71/guia_elaboracion_proyectos_c.pdf>

Artículos web:

[18] Steve Crowe. *ABB acquiring AB Rotech to bolster automotive robotic welding. The Robot Report* [en línea]. 11 julio 2018, [Consulta: 16 mayo 2019]. Disponible a: <https://www.therobotreport.com/abb-acquiring-ab-rotech-robotic-welding/>>.

Estudios estadísticos:

[19] Dirección General de Estadística y Análisis Sociolaboral. *Estadística de Accidentes de Trabajo*. Resumen de principales resultados [en línea]. 2006, [Consulta: 22 mayo 2019]. Disponible a: http://www.mitramiss.gob.es/estadisticas/eat/eat16/Resumen_resultados_ATR_2016.pdf.

Páginas web:

[20] SIEMENS. *Potencia + eficiencia: SIMATIC S7-1500 plus TIA Portal*. [en línea]. SIEMENS, 2019 [Consulta: 12 abril 2019]. Disponible a: https://w5.siemens.com/spain/web/es/industry/automatizacion/simatic/controladores_modulares/controlador_avanzado/s71500/pages/default.aspx

Video tutoriales:

[21] ABB. *Getting started tutorials for RobotStudio*. [en línea]. ABB, 2019 [Consulta: 2 febrero 2019]. Disponible a: <https://new.abb.com/products/robotics/robotstudio/how-to-use-it/getting-started>